XIX Conferencia de la Asociación Española para la Inteligencia Artificial (CAEPIA 20/21)

Septiembre 22-24

Málaga, España



E. Alba, F. Chicano, G. Luque, R. Gil-Merino, C. Cotta, D. Camacho, M. Ojeda-Aciego, S. Montes, A. Troncoso, J. Riquelme, E. Onaindia, M.J. del Jesús, J.A. Gámez, A. Bugarín, M. Marcos, A. Ledezma, J.P. Llerena, J. Echanobe, J. Toutouh, S. Muiños (Eds.)

XIX Conferencia de la Asociación Española para la Inteligencia Artificial CAEPIA 20/21

22-24 de septiembre de 2021

Málaga, España

Enrique Alba, Francisco Chicano, Gabriel Luque, Rodrigo Gil-Merino, Carlos Cotta, David Camacho, Manuel Ojeda-Aciego, Susana Montes, Alicia Troncoso, José Riquelme, Eva Onaindia, María José del Jesús, José Antonio Gámez, Alberto Bugarín, Mar Marcos, Agapito Ledezma, Juan Pedro Llerena, Javier Echanobe, Jamal Toutouh, Santiago Muiños



Editores

Enrique Alba
Universidad de Málaga
Málaga, España

Francisco Chicano
Universidad de Málaga
Málaga, España

Gabriel Luque
Universidad de Málaga
Málaga, España

Rodrigo Gil-Merino
Universidad de Málaga
Málaga, España

Carlos Cotta
Universidad de Málaga
Málaga, España

David Camacho
Universidad Politécnica de Madrid
Madrid, España

Manuel Ojeda-Aciego
Universidad de Málaga
Málaga, España

Susana Montes
Universidad de Oviedo
Oviedo, España

Alicia Troncoso
Universidad Pablo de Olavide
Sevilla, España

José Riquelme
Universidad de Sevilla
Sevilla, España

Eva Onaindia
Universidad Politécnica de Valencia
Valencia, España

María José del Jesús
Universidad de Jaén
Jaén, España

José Antonio Gámez
Universidad de Castilla La Mancha
Albacete, España

Alberto Bugarín
Universidad de Santiago de Compostela
Santiago de Compostela, España

Mar Marcos
Universitat Jaume I
Castellón de la Plana, España

Agapito Ledezma
Universidad Carlos III de Madrid
Madrid, España

Juan Pedro Llerena
Universidad Carlos III de Madrid
Madrid, España

Javier Echanobe
Universidad del País Vasco
Bilbao, España

Jamal Toutouh
Universidad de Málaga
Málaga, España

Santiago Muiños
Centro Tecnológico AIMEN
Pontevedra, España

ISBN: 978-84-09-30514-8
© Los autores, 2021

Presentación de CAEPIA 20/21

Este volumen contiene un conjunto de artículos seleccionados y revisados por pares enviados a CAEPIA 20/21, la XIX Conferencia de la Asociación Española de Inteligencia Artificial, celebrada en Málaga, España, del 22 al 24 de septiembre de 2021. CAEPIA es un evento bienal español bien establecido sobre Inteligencia Artificial (IA) que comenzó en 1985. Ediciones anteriores tuvieron lugar en Alicante, Málaga, Murcia, Gijón, San Sebastián, Santiago de Compostela, Sevilla, La Laguna, Madrid, Albacete, Salamanca y Granada.

CAEPIA es un foro nacional abierto a investigadores de todo el mundo para presentar y discutir sus últimos avances científicos y tecnológicos en IA. Los autores podían optar por cinco tipos de contribuciones: trabajos inéditos de investigación para un volumen en la serie *Lecture Notes in Artificial Intelligence* de Springer, trabajos inéditos de investigación para estas actas, trabajos destacados ya publicados, proyectos de doctorado, desarrollos de aplicaciones móviles y vídeos divulgativos. La conferencia acogió tanto investigación teórica como metodológica, técnica y aplicada.

Dentro de CAEPIA se organizaron varios talleres y congresos federados relacionados con los temas más relevantes de la IA: XX Congreso Español Sobre Tecnologías y Lógica Fuzzy (ESTYLF); XIV Congreso Español de Metaheurísticas, Algoritmos Evolutivos y Bioinspirados (MAEB); X Simposio de Teoría y Aplicaciones de la Minería de Datos (TAMIDA); y seis talleres. También contamos con un Doctoral Consortium (DC). Este es un foro para que los estudiantes de doctorado interactúen con otros investigadores discutiendo sus planes de trabajo y avances en el doctorado. Como actividad adicional de IA, llevamos a cabo el 4º Concurso de Aplicaciones Móviles con Técnicas de IA, junto con una nueva edición del Concurso de Vídeos de Divulgación de IA.

Todas las actividades anteriores avalan la IA, y nos esforzamos por alcanzar una alta calidad en los artículos científicos, el DC y las competiciones. El programa científico de CAEPIA 20/21 también ofreció una vía para difundir trabajos destacados (Key Works: KW) publicados recientemente en revistas y foros de alto impacto científico. CAEPIA siempre ha tenido como objetivo ser reconocida como una conferencia insignia en IA y, por lo tanto, los artículos fueron revisados por pares. El número total de envíos a CAEPIA 20/21 fue de 186 (en este número no se incluyeron ni DC ni concursos ni presentaciones KW, que suman 83 contribuciones adicionales, y que pasaron por un proceso de evaluación diferente). Los revisores evaluaron la calidad general de los manuscritos presentados, junto con la calidad de la metodología empleada, la solidez de las conclusiones, la importancia del tema, la claridad de la redacción y su organización, entre otros criterios de evaluación. A partir de estas revisiones, los responsables de área, presidentes de congresos y organizadores de talleres y sesiones especiales propusieron un número final de artículos que fueron analizados y aprobados por los editores de este volumen.

CAEPIA 20/21 invitó a dos investigadores de renombre internacional a impartir una charla plenaria. Nuestros dos ponentes plenarios fueron Óscar Cordón (Inteligencia Artificial para Antropología Forense e Identificación Humana) y Yaochu Jin (Optimización Evolutiva Basada en Datos). Nuestra conferencia se celebró como un gran evento dentro de uno aún mayor: la Conferencia Española de Informática (CEDI), que también contó con charlas plenarias muy interesantes.

AEPIA y los organizadores de CAEPIA 20/21 reconocieron las mejores tesis doctorales y artículos originales en eventos federados escritos tanto por investigadores consolidados como por estudiantes. CAEPIA 20/21 también tuvo como objetivo promover la presencia de mujeres en la investigación de IA. Como en ediciones anteriores, el premio Frances Allen reconoció las dos mejores tesis doctorales defendidas por una mujer durante los dos últimos años.

Los editores de este volumen quieren agradecer a las numerosas personas que contribuyeron al éxito de CAEPIA 20/21: autores, miembros de los comités científicos y los comités de programa, ponentes invitados, organizadores de eventos, gestores de medios electrónicos, etc. También, agradecer el trabajo incansable del comité organizador, nuestros patrocinadores (como VRAIN en Valencia), el equipo de Springer y AEPIA por su apoyo.

Por último, pero no menos importante, en nombre de los participantes de CAEPIA 20/21, Enrique Alba (presidente) y Francisco Chicano (responsable de este volumen) dan las gracias a la organización de CEDI, la Universidad de Málaga (sede local de la conferencia) y a toda la comunidad española que trabaja en IA (y sus numerosos colaboradores extranjeros) por hacer de este evento un verdadero éxito.

Enrique Alba
Presidente de CAEPIA 20/21

Presentación de la Presidenta de la Asociación Española para la Inteligencia Artificial

Es para mí un inmenso honor presentaros esta nueva edición en Málaga de la serie de conferencias CAEPIA, en la que además se cierra una etapa en la que he tenido el inmenso privilegio de trabajar como Presidenta de la Asociación Española para la Inteligencia Artificial (AEPIA).

Debemos el inicio de estas conferencias bienales en los años 80 a un pequeño, pero muy entusiasta, grupo de investigadores españoles en Inteligencia Artificial (IA). A pesar de que en aquella época la IA estaba en un momento álgido, la situación no era comparable al momento actual, en el que nuestra disciplina se ha convertido en una de las más influyentes en la nueva revolución tecnológica que está teniendo lugar. Además, hay ciertas diferencias con entonces que auguran que quizás, y aunque quede mucho camino aún por recorrer, esta vez la primavera puede ser casi eterna. Hay algunos factores que han confluido para dar este estallido de vida a la IA, como son la disponibilidad de enormes cantidades de datos (uno de los alimentos de los algoritmos actuales de IA), el abaratamiento de la computación en nube o la existencia de plataformas distribuidas y paralelas, que permiten el procesado rápido y económicamente viable de esas grandes cantidades de datos. Confluye también un imparable cambio social, ya que todos interactuamos constantemente con nuestros múltiples dispositivos móviles, en un mundo en el que la interconectividad es imprescindible, y aún lo es más desde la irrupción de la pandemia CoVid-19 en nuestras vidas.

Nuestra disciplina es la cabeza del cambio, y ello implica no sólo cambios tecnológicos y sociales, sino también de poder económico y geopolítico. Existe una cierta dinámica de confrontación entre EEUU y China, en la que esta última ha explicitado que su meta es convertirse en la nación que domine el mundo en IA en el año 2030, mediante una inversión masiva en I+D+I y en políticas de fusión "militar-civil", aunque por el momento son las empresas americanas las que van en cabeza. Y en este contexto, la UE se ha decidido por una tercera vía, la de intentar fomentar y mantener su talento investigador en IA, además de atraer talento extranjero, y promover asimismo una IA robusta, transparente, auditable, ética y al servicio de la humanidad, con las personas en el centro. Los diferentes países de la UE, entre ellos España, han elaborado Estrategias de Inteligencia Artificial. Según estudios de varias consultoras, y en palabras del Presidente del Gobierno en la presentación de la Estrategia Nacional de IA (ENIA) recientemente este 2 de Diciembre, la IA ha contribuido en 2018 con unos 1700 millones de euros al PIB mundial, y en 2030 se estima que aportará ocho veces más, alrededor de 14 mil millones de euros, una cifra cercana a todo el PIB actual de la UE. Es indudable que la disciplina tiene una enorme capacidad de transformación en la economía.

Nuestra ENIA es más modesta que las estrategias de otros países europeos en inversión, así que es mandatorio para hacer valer nuestras fortalezas como país. Y es en este esfuerzo colectivo donde contribuimos los científicos que trabajamos en los diferentes campos de la IA. En esta conferencia estarán representados muchos de los grupos españoles más relevantes, aportando nuestra investigación y nuestra capacidad de transferencia, parte de la cual se plasma en los trabajos que se presentan tanto en las conferencias plenarias como en las diferentes sesiones, Key notes, Workshops, etc. incluidos en esta XIX CAEPIA, que es co-ocurrente con las conferencias del XX Congreso Español de Tecnologías y Lógica Difusa (ESTYLF), el XIV Congreso Español de Metaheurística y Algoritmos evolutivos y bioinspirados (MAEB), y el X Simposium de Teoría y Aplicaciones de Minería de Datos (TAMIDA), todos ellos co-celebrándose además junto con el Congreso Español De Informática (CEDI). La conferencia también se implica con el talento emergente, incluyendo programas para los estudiantes y los jóvenes investigadores, como es el caso del Doctoral Consortium, los Premios a los Jóvenes investigadores autores de los mejores artículos, o a las mejores Apps y videos divulgativos de IA. En AEPIA también contribuimos con acciones de género positivas mediante los premios Frances Allen a las mejores tesis doctorales realizadas por mujeres, en un intento de reducir la significativa brecha de género e incorporar más talento a la disciplina, en esta ocasión agradecemos su patrocinio en este Premio al Instituto Valenciano de Investigación en Inteligencia Artificial (VRAIN).

Tenemos un tejido investigador del que sentirnos orgullosos, debemos trabajar para mantener, retener y atraer talento, una de las materias primas fundamentales para abrazar el futuro. En esta XIX CAEPIA aportamos una muestra de nuestra contribución a esa apuesta de futuro.

Amparo Alonso Betanzos
Presidenta de AEPIA

Ponentes Plenarios

Óscar Cordón
Instituto Andaluz Interuniversitario en Ciencia de Datos e Inteligencia Computacional (Instituto DaSCI)



Oscar Cordón fue el Director Fundador del Centro de Enseñanzas Virtuales (2001-05) y Delegado de la Rectora para la Universidad Digital (2015-19) de la Universidad de Granada (UGR). Fue uno de los investigadores fundadores del European Centre for Soft Computing (2006-11), siendo luego contratado como Investigador Afiliado Distinguido hasta diciembre de 2015. En la actualidad es Catedrático de Universidad en la UGR. Durante más de 25 años, ha impulsado programas de investigación y transferencia en fundamentos y aplicaciones de inteligencia computacional con un gran reconocimiento internacional. Ha publicado más de 380 contribuciones científicas, incluyendo un libro de investigación sobre Genetic Fuzzy Systems (con más de 1400 citas en Google Scholar) y 112 artículos de revista JCR-SCI (68 en Q1 y 38 en D1), ha dirigido 19 tesis doctorales y coordinado 37 proyectos y contratos de investigación (con un presupuesto global de más de 9M€). A fecha de mayo de 2021, sus publicaciones han recibido 5422 citas (H-index=39), estando incluido en el 1 % de investigadores más citados en el mundo (fuente: Web of Science); con 14687 citas y H-index=58 en Google Scholar. También tiene una patente internacional en explotación sobre un sistema inteligente para identificación forense, comercializada en México y Sudáfrica.

Ha recibido el Premio de Jóvenes Investigadores de la UGR (2004), el Premio IEEE CIS Outstanding Early Career Award (en su primera edición, 2011), el Premio Nacional de Informática ARITMEL (2014) por la Sociedad Científica de España, el IEEE Fellow (2018) y el IFSA Fellow (2019). Fue miembro del Grupo de Expertos que desarrolló la Estrategia Española de I+D+I en Inteligencia Artificial para el Ministerio de Ciencia, Innovación y Universidades (2018-19). Es o ha sido Editor Asociado de 19 revistas internacionales, siendo reconocido como Outstanding Associate Editor de IEEE Transactions on Fuzzy Systems (2008) y de IEEE Transactions on Evolutionary Computation (2019). Desde 2004, ha ocupado distintos puestos de representación en EUSFLAT e IEEE Computational Intelligence Society.

En la actualidad investiga en inteligencia artificial para identificación forense (en colaboración con el laboratorio de Antropología Física de la UGR y varios laboratorios forenses y cuerpos y fuerzas de seguridad internacionales) y en modelado basado en agentes y análisis de redes sociales para marketing (en colaboración con R0D Brand Consultants en proyectos para CAPSA, Mercedes, Jaguar-Land Rover, El Corte Inglés, Telefónica, Samsung, Coca Cola Europa, Cola Cao, WiZink,...).

Inteligencia Artificial para Antropología Forense e Identificación Humana

Los métodos de identificación forense basados en el esqueleto empleados por antropólogos, odontólogos y patólogos representan el primer paso en cualquier proceso de identificación humana (ID) y la última oportunidad de identificación de la víctima cuando no puede aplicarse el análisis de ADN o de huellas dactilares. Incluyen métodos como la estimación del perfil biológico (BP), la radiografía comparativa (CR), la superposición craneofacial (CFS) y la comparación de registros dentales. La BP implica el estudio de restos óseos para encontrar rasgos característicos (edad, sexo, estatura y ascendencia) que ayuden a determinar la identidad del individuo. Desempeña un papel crucial en la reducción del rango de coincidencias potenciales durante el proceso de ID, antes de la confirmación mediante una o más técnicas de ID. La CR considera la comparación ante-mortem (AM) y post-mortem (PM) de diferentes huesos y cavidades (senos frontales del cráneo, clavículas, rótulas,...) que han demostrado ser útiles para la identificación positiva por su individualidad y singularidad. La CFS tiene como objetivo superponer un cráneo con algunas imágenes AM de un candidato para determinar si corresponden a la misma persona.

Sin embargo, los profesionales todavía emplean un paradigma de observación utilizando métodos subjetivos introducidos hace muchas décadas basados en la descripción oral y documentación escrita de los hallazgos obtenidos y la comparación manual y visual de los datos AM y PM. El diseño de métodos sistemáticos, automáticos y confiables para apoyar al antropólogo forense en la aplicación de BP, CFS y CR, evitando el uso de procedimientos manuales subjetivos, propensos a errores y que consumen mucho tiempo, es una necesidad para mejorar la identificación forense. El uso de inteligencia artificial, en particular inteligencia computacional (algoritmos evolutivos, conjuntos difusos y aprendizaje profundo), visión por ordenador (registrado de imágenes 3D-2D y procesamiento de imágenes) y aprendizaje automático explicable es una forma muy adecuada para lograr este objetivo. En esta charla presentaremos tres sistemas inteligentes para CFS, CR y estimación de la edad de la muerte a partir del esqueleto, desarrollados en colaboración con el Laboratorio de Antropología Física de la Universidad de Granada en el marco de un proyecto de investigación desarrollado en los últimos quince años. Uno de estos sistemas está protegido por una patente internacional explotada por Panacea Cooperative Research y está comercializado en diferentes países.

Yaochu Jin
Universidad de Surrey, Reino Unido



Yaochu Jin se licenció, estudió y se doctoró en la Universidad de Zhejiang (Hangzhou, China) en 1988, 1991 y 1996, respectivamente, y obtuvo el título de ingeniero de la Universidad del Ruhr (Bochum, Alemania) en 2001.

En la actualidad es profesor distinguido de Inteligencia Computacional en el Departamento de Informática de la Universidad de Surrey, Guildford, Reino Unido, donde dirige el Grupo de Computación e Ingeniería Inspirada en la Naturaleza. Ha sido "Finland Distinguished Professor"de la Universidad de Jyvaskyla (Finlandia), "Changjiang Distinguished Visiting Professor"de la Universidad de Northeastern (China) y "Distinguished Visiting Scholar"de la Universidad Tecnológica de Sydney (Australia). Recientemente, el Ministerio Federal de Educación e Investigación de Alemania le ha concedido la "Cátedra Alexander von Humboldt de Inteligencia Artificial". Sus principales intereses de investigación son la optimización evolutiva asistida por datos, el aprendizaje automático fiable, el aprendizaje evolutivo multiobjetivo, la robótica de enjambre y los sistemas evolutivos de desarrollo.

El Dr. Jin es actualmente editor jefe de IEEE Transactions on Cognitive and Developmental Systems y de Complex & Intelligent Systems. Fue conferenciante distinguido del IEEE y vicepresidente de la Sociedad de Inteligencia Computacional del IEEE. Ha recibido el premio al mejor artículo de 2018 y 2020 de IEEE Transactions on Evolutionary Computation, el premio al mejor artículo de 2014, 2016 y 2019 de IEEE Computational Intelligence Magazine y el premio al mejor artículo del Simposio de IEEE sobre Inteligencia Computacional en Bioinformática y Biología Computacional de 2010. Está reconocido como Investigador Altamente Citado en 2019 y 2020 por el Grupo Web of Science. Es miembro de IEEE.

Optimización Evolutiva Basada en Datos

Muchos problemas de optimización del mundo real no tienen funciones objetivo analíticas y las evaluaciones de los objetivos deben basarse en costosos cálculos o experimentos físicos. Estos problemas de optimización se conocen como problemas de optimización basados en datos. Esta charla ofrece una visión general de la optimización evolutiva asistida por datos de sistemas complejos. Comenzamos con una breve introducción a las ideas básicas de la optimización evolutiva basada en datos, seguida de estrategias avanzadas de gestión de sustitutos que hacen uso de técnicas avanzadas de aprendizaje automático como el aprendizaje semisupervisado, el aprendizaje de transferencia y el aprendizaje de conjunto. Se expondrán ejemplos del mundo real, desde la optimización del diseño de ingeniería hasta la búsqueda de arquitecturas neuronales.

Organización

Presidente

Enrique Alba	Universidad de Málaga, España

Presidenta del Comité de Premios

Eva Onaindia	Universidad Politécnica de Valencia, España

Presidentes de Talleres y Trabajos Destacados

Rodrigo Gil-Merino	Universidad de Málaga, España
Araceli Sanchís	Universidad Carlos III de Madrid, España

Presidente de la Sesión General de CAEPIA

Francisco Chicano	Universidad de Málaga, España

Presidentes de Competiciones

Alberto Bugarín	Universidad de Santiago de Compostela, España
José Antonio Gámez	Universidad de Castilla La Mancha, España

Responsable de Actas LNAI

Gabriel Luque	Universidad de Málaga, España

Presidentes del Doctoral Consortium

José Riquelme	Universidad de Sevilla, España
María José Del Jesús Díaz	Universidad de Jaén, España

Responsables de Área de CAEPIA

Óscar Corcho	Universidad Politécnica de Madrid, España
Javier del Ser	Tecnalia, España
Juan Manuel Fernández Luna	Universidad de Granada, España
Jesús García Herrero	Universidad Carlos III de Madrid, España
Pedro Larrañaga	Universidad Politécnica de Madrid, España
Patricio Martínez Barco	Universidad de Alicante, España
Miguel Molina Solana	Universidad de Granada, España
Serafín Moral Callejón	Universidad de Granada, España
Eva Onaindia	Universidad Politécnica de Valencia, España
Juan Pavón Mestras	Universidad Complutense de Madrid, España
David Pearce	Universidad Politécnica de Madrid, España
José Luis Pérez de la Cruz Molina	Universidad de Málaga, España
Petia Radeva	Universtat de Barcelona & Centro de Visión por Computador, España

Presidentes de Conferencias Federadas

XX Congreso Español sobre Tecnologías y Lógica Fuzzy (ESTYLF)

Manuel Ojeda-Aciego	Universidad de Málaga, España
Susana Montes	Universidad de Oviedo, España

XIV Congreso Español de Metaheurísticas, Algoritmos Evolutivos y Bioinspirados (MAEB)

Carlos Cotta	Universidad de Málaga, España
David Camacho	Universidad Politécnica de Madrid, España

X Simposio de Teoría y Aplicaciones de Minería de Datos (TAMIDA)

Alicia Troncoso	Universidad Pablo de Olavide, España
José Riquelme	Universidad de Sevilla, España

Organizadores de Talleres

II Taller de Grupos de investigación españoles de IA en Biomedicina (IABiomed)

Mar Marcos	Universitat Jaume I, España
David Riaño	Universitat Rovira i Virgili, España
José M. Juárez	Universidad de Murcia, España

I Taller de La IA aplicada a los Sistemas Inteligentes de Transporte (IA-SIT)

Paz Sesmero	Universidad Carlos III de Madrid, España
Pablo Marín	Universidad Carlos III de Madrid, España
Juan Pedro Llerena	Universidad Carlos III de Madrid, España

I Taller de Inteligencia Artificial para el Desarrollo (AI4D)

Agapito Ledezma Espino	Universidad Carlos III de Madrid, España
José Antonio Iglesias Martínez	Universidad Carlos III de Madrid, España
David Camilo Corrales Muñoz	Institut national de la recherche agronomique, Francia
Elias Said Hung	Universidad Internacional de La Rioja, España

I Taller de Inteligencia Artificial Cuántica (QUARTI)

Javier Echanobe	Universidad del País Vasco, España
Enrique Alba	Universidad de Málaga, España

I Taller sobre Modelos Generativos Profundos Aplicados (AGM)

Carlos González Val	Centro Tecnológico AIMEN, España
Juan Manuel Fernández Montenegro	Centro Tecnológico AIMEN, España
Andrea Fenández Martinez	Centro Tecnológico AIMEN, España
Santiago Muiños Landin	Centro Tecnológico AIMEN, España
Omar A. Mures	Universidad de A Coruña (UDC), España
José A. Iglesias-Guitián	Universidad de A Coruña (UDC), CITIC, España

I Taller Ciudades inteligentes e Inteligencia Artificial (SCAI)

Jamal Toutouh	Universidad de Málaga, España
Enrique Alba	Universidad de Málaga, España

Organizadores de Sesiones Especiales

ESTYLF. Toma de decisiones con información difusa

Rosa Rodríguez	Universidad de Jaén, España
Álvaro Labella	Universidad de Jaén, España
Luis Martínez	Universidad de Jaén, España

ESTYLF. Funciones de agregación y preagregación y sus aplicaciones

Humberto Bustince	Universidad Pública de Navarra, España
Graçaliz Dimuro	Universidad Pública de Navarra, España
Javier Fernandez	Universidad Pública de Navarra, España

MAEB. Algoritmos Bioinspirados Profundos

Francisco Chávez	Universidad de Extremadura, España
Pablo García	Universidad de Granada, España

MAEB. Aplicaciones en Medicina y Biotecnología

Ignacio Hidalgo	Universidad Complutense de Madrid, España
José Manuel Velasco	Universidad Complutense de Madrid, España

MAEB. Estrategias de implementación eficiente de heurísticas y metaheurística

José Manuel Colmenar	Universidad Rey Juan Carlos, España
Eduardo García Pardo	Universidad Rey Juan Carlos, España

MAEB. Metaheurísticas Aplicadas al Análisis de Redes Sociales

Jesús Sánchez-Oro	Universidad Rey Juan Carlos, España
Antonio González-Pardo	Universidad Rey Juan Carlos, España
Abraham Duarte	Universidad Rey Juan Carlos, España

Comité de Organización

Publicidad

Rubén Saborido	Universidad de Málaga, España

Diseño Gráfico

Christian Cintrano	Universidad de Málaga, España
José Ángel Morell	Universidad de Málaga, España

Web

José Ángel Morell	Universidad de Málaga, España

Responsables IT

Francisco Chicano	Universidad de Málaga, España
Andrés Camero	Centro Aeroespacial Alemán, Alemania

Equipo de Soporte IT

Zakaria Abdelmoiz DAHI	Universidad de Málaga, España
Rubén Saborido	Universidad de Málaga, España
Christian Cintrano	Universidad de Málaga, España

Comités de Programa

Sesión General de CAEPIA

Aghaei, Maedeh	Universidad de Barcelona, España
Alonso-Betanzos, Amparo	Universidad de La Coruña, España
Bacardit, Jaume	Newcastle University, Reino Unido
Bahamonde, Antonio	Universidad de Oviedo, España
Bajo, Javier	Universidad Politécnica de Madrid, España
Balaguer-Ballester, Emili	Bournemouth University, Reino Unido
Barbancho, Ana M.	Universidad de Málaga, España
Barreiro, Alvaro	Universidad de La Coruña, España
Barrenechea, Edurne	Universidad Pública de Navarra, España
Barro, Senén	Universidad de Santiago de Compostela, España
Bañares, Jose Angel	Universidad de Zaragoza, España
Bernardos, Ana M.	Universidad Politécnica de Madrid, España
Bielza Lozoya, Concha	Universidad Politécnica de Madrid, España
Borrajo, Daniel	Universidad Carlos III de Madrid, España
Botia, Juan	King's College London, Reino Unido
Botti, Vicent	Universitat Politècnica de València, España
Bugarín, Alberto	Universidad de Santiago de Compostela, España
Bustince, Humberto	Universidad Pública de Navarra, España
Cadenas, José M.	Universidad de Murcia., España
Callejas, Zoraida	Universidad de Granada, España
Cantador, Iván	Universidad Autónoma de Madrid, España
Castelo, Robert	Universitat Pompeu Fabra, España
Corchuelo, Rafael	Universidad de Sevilla, España
Cordon, Oscar	Universidad de Granada, España
Damas, Sergio	Universidad de Granada, España
de Campos, Luis M.	Universidad de Granada, España
de Carvalho, Andre	Universidad de São Paulo, Brasil
De La Ossa, Luis	Universidad de Castilla-La Mancha, España
Del Coz, Juan José	Universidad de Oviedo, España
Dorronsoro, Jose	Universidad Autónoma de Madrid, España
Díez, Francisco Javier	Universidad Nacional de Educación a Distancia, España
Fernández López, Mariano	Universidad San Pablo CEU, España
Fernández-Caballero, Antonio	Universidad de Castilla-La Mancha, España
Ferri, Francesc J.	Universidad de Valencia, España
Galán, José Manuel	Universidad de Burgos, España
Gamez, Jose	Universidad de Castilla-La Mancha, España
García Bringas, Pablo	Universidad de Deusto, España
García-Castro, Raúl	Universidad Politécnica de Madrid, España
Gasca, Rafael M.	Universidad de Sevilla, España
Godo, Lluis	IIIA - CSIC, España
Gomez Romero, Juan	Universidad de Granada, España
Gonzalez, Antonio	Universidad de Granada, España
Graña, Manuel	Universidad del País Vasco, España
Guijarro-Berdiñas, Bertha	Universidad de La Coruña, España
Gómez-Rodríguez, Carlos	Universidade da Coruña, España
Hervas, Cesar	Universidad de Córdoba, España
Inza, Inaki	Universidad del País Vasco, España
Jimenez Linares, Luis	Universidad de Castilla-La Mancha, España
Levy, Jordi	IIIA - CSIC, España
Lopez, Adolfo	Universidad de Valladolid, España
Luaces, Oscar	Universidad de Oviedo, España
López, Victoria	CUNEF Universidad, España
Mandow, Lawrence	Universidad de Málaga, España

Manyà, Felip	IIIA-CSIC, España
Martínez, Luis	Universidad de Jaén, España
Martinez Tomas, Rafael	Universidad Nacional de Educación a Distancia, España
Massa, Jose	Universidad de Buenos Aires, Argentina
Medina-Carnicer, Rafael	Universidad de Córdoba, España
Melian, Belen	Universidad de La Laguna, España
Meseguer, Pedro	IIIA - CSIC, España
Molina, Jose M.	Universidad Carlos III de Madrid, España
Morales-Bueno, Rafael	Universidad de Málaga, España
Muguerza, Javier	Universidad del País Vasco, España
Navas, Ismael	Universidad de Málaga, España
Olivas, Jose Angel	Universidad de Castilla-La Mancha, España
Ossowski, Sascha	Universidad Rey Juan Carlos, España
Palma, Jose	Universidad de Murcia, España
Parraga, C. Alejandro	Centro de Visión por Computador, España
Parreño, Francisco	Universidad de Castilla-La Mancha, España
Patricio, Miguel Angel	Universidad Carlos III de Madrid, España
Peregrin, Antonio	Universidad de Huelva, España
Pla, Filiberto	Universidad Jaume I, España
Pomares, Hector	Universidad de Granada, España
Puerta, Jose M	Universidad de Castilla-La Mancha, España
R. Vela, Camino	Universidad de Oviedo, España
Rizo, Ramon	Universidad de Alicante, España
Rosso, Paolo	Universitat Politècnica de València, España
Salmeron, Antonio	Universidad de Almería, España
Sanchez, Jose Salvador	Universitat Jaume I, España
Sanchez, Luciano	Universidad de Oviedo, España
Saquete, Estela	Universidad de Alicante, España
Segarra, Encarna	Universitat Politècnica de València, España
Soria, Emilio	Universidad de Valencia, España
Taboada, Maria	Universidad de Santiago de Compostela, España
Talavera, Estefania	Universidad de Delft, Países Bajos
Ureña-López, L. Alfonso	Universidad de Jaén, España
Valencia-Garcia, Rafael	Universidad de Murcia, España
Ventura, Sebastián	Universidad de Córdoba, España
Verdegay, José Luis	Universidad de Granada, España

Volumen de LNAI

Aghaei Gavari, Maedeh	Universidad de Barcelona, España
Alonso-Betanzos, Amparo	Universidad de La Coruña, España
Bacardit, Jaume	Newcastle University, Reino Unido
Bahamonde, Antonio	Universidad de Oviedo, España
Bajo, Javier	Universidad Politécnica de Madrid, España
Balaguer-Ballester, Emili	Bournemouth University, Reino Unido
Barbancho, Ana M.	Universidad de Málaga, España
Barreiro, Alvaro	Universidad de La Coruña, España
Barrenechea, Edurne	Universidad Pública de Navarra, España
Barro, Senén	Universidad de Santiago de Compostela, España
Bañares, Jose Angel	Universidad de Zaragoza, España
Bernardos, Ana M.	Universidad Politécnica de Madrid, España
Bielza Lozoya, Concha	Universidad Politécnica de Madrid, España
Borrajo, Daniel	Universidad Carlos III de Madrid, España
Botia, Juan	King's College London, Reino Unido
Botti, Vicent	Universitat Politècnica de València, España
Bustince, Humberto	Universidad Pública de Navarra, España
Cabalar, Pedro	Universidad de La Coruña, España
Cadenas, José M.	Universidad de Murcia, España
Callejas, Zoraida	Universidad de Granada, España
Cantador, Iván	Universidad Autónoma de Madrid, España

Castelo, Robert	Universitat Pompeu Fabra, España
Corchuelo, Rafael	Universidad de Sevilla, España
Damas, Sergio	Universidad de Granada, España
de Campos, Luis M.	Universidad de Granada, España
de Carvalho, Andre	Universidad de São Paulo, Brazil
De La Ossa, Luis	Universidad de Castilla-La Mancha, España
Del Coz, Juan José	Universidad de Oviedo, España
Diaz-Rodriguez, Natalia	ENSTA Paris Institut Polytechnique Paris, France
Dorronsoro, Jose	Universidad Autónoma de Madrid, España
Fandinno, Jorge	Potsdam Universty, Alemania
Fernández López, Mariano	Universidad San Pablo CEU, España
Fernández-Caballero, Antonio	Universidad de Castilla-La Mancha, España
Ferri, Francesc J.	Universidad de Valencia, España
Galán, José Manuel	Universidad de Burgos, España
Gamez, Jose	Universidad de Castilla-La Mancha, España
García-Castro, Raúl	Universidad Politécnica de Madrid, España
Gasca, Rafael M.	Universidad de Sevilla, España
Godo, Lluis	IIIA - CSIC, España
Gomez Romero, Juan	Universidad de Granada, España
Gonzalez, Antonio	Universidad de Granada, España
Graña, Manuel	Universidad del País Vasco, España
Gómez-Rodríguez, Carlos	Universidade da Coruña, España
Hervas, Cesar	Universidad de Córdoba, España
Inza, Inaki	Universidad del País Vasco, España
Jimenez Linares, Luis	Universidad de Castilla-La Mancha, España
Kushibar, Kaisar	Universidad de Barcelona, España
Levy, Jordi	IIIA - CSIC, España
Lopez, Adolfo	Universidad de Valladolid, España
López, Victoria	CUNEF Universidad, España
Mandow, Lawrence	Universidad de Málaga, España
Manyà, Felip	IIIA-CSIC, España
Martinez-Camara, Eugenio	Universidad de Granada, España
Martínez, Luis	Universidad de Jaén, España
Massa, Jose	Universidad de Buenos Aires, Argentina
Medina-Carnicer, Rafael	Universidad de Córdoba, España
Mejia, Marcos	Universitat de Barcelona, España
Melian, Belen	Universidad de La Laguna, España
Meseguer, Pedro	IIIA - CSIC, España
Millan, Eva	Universidad de Málaga, España
Molina, Jose M.	Universidad Carlos III de Madrid, España
Morales-Bueno, Rafael	Universidad de Málaga, España
Nagarajan, Bhalaji	Universitat de Barcelona, España
Nuñez, Jose Fernando	Universitat de Barcelona, España
Olivas, Jose Angel	Universidad de Castilla-La Mancha, España
Ossowski, Sascha	Universidad Rey Juan Carlos, España
Parreño, Francisco	Universidad de Castilla-La Mancha, España
Patricio, Miguel Angel	Universidad Carlos III de Madrid, España
Peregrin, Antonio	Universidad de Huelva, España
Pezzano, Giuseppe	Universitat de Barcelona, España
Pla, Filiberto	Universidad Jaume I, España
Pomares, Hector	Universidad de Granada, España
Puerta, Jose M	Universidad de Castilla-La Mancha, España
R. Vela, Camino	Universidad de Oviedo, España
Rizo, Ramon	Universidad de Alicante, España
Rodenas, Javier	Universidad de Barcelona, España
Rodríguez, Rosa	Universidad de Jaén, España
Rosso, Paolo	Universitat Politècnica de València, España
Salmeron, Antonio	Universidad de Almería, España
Sanchez, Jose Salvador	Universitat Jaume I, España

Sanchez, Luciano	Universidad de Oviedo, España
Saquete, Estela	Universidad de Alicante, España
Segarra, Encarna	Universitat Politècnica de València, España
Soria, Emilio	Universidad de Valencia, España
Talavera, Estefania	Universidad de Groningen, Países Bajos
Ureña-López, L. Alfonso	Universidad de Jaén, España
Valencia-Garcia, Rafael	Universidad de Murcia, España
Ventura, Sebastián	Universidad de Córdoba, España
Verdegay, José Luis	Universidad de Granada, España
Zaied, Mourad	Universidad de Gabes, Túnez

ESTYLF

Alcala, Rafael	Universidad de Granada, España
Alcala-Fdez, Jesus	Universidad de Granada, España
Alcalde, Cristina	Universidad del País Vasco, España
Alonso, Jose M.	Universidad de Santiago de Compostela, España
Barrenechea, Edurne	Universidad Pública de Navarra, España
Barro, Senén	Universidad de Santiago de Compostela, España
Benítez, José M.	Universidad de Granada, España
Bobillo, Fernando	Universidad de Zaragoza, España
Bugarín, Alberto	Universidade de Santiago de Compostela, España
Burusco, Ana	Universidad Pública de Navarra, España
Bustince, Humberto	Universidad Pública de Navarra, España
Cabrera, Inma P.	Universidad de Málaga, España
Cabrerizo, Francisco Javier	Universidad de Granada, España
Calvo, Tomasa	Universidad de Alcalá, España
Carmona, Pablo	Universidad de Extremadura, España
Castro-Schez, Jose Jesus	Universidad de Castilla-La Mancha, España
Cordero, Pablo	Universidad de Málaga, España
Cornejo Piñero, Maria Eugenia	Universidad de Cádiz, España
Couso, Inés	Universidad de Oviedo, España
Cubillo, Susana	Universidad Politécnica de Madrid, España
De Andres, Rocio	Universidad de Salamanca, España
De Soto, Adolfo R.	Universidad de León, España
Del Jesus, Maria Jose	Universidad de Jaén, España
Díaz, Susana	Universidad de Oviedo, España
Elorza, Jorge	Universidad de Navarra, España
Escaño, Juan Manuel	Universidad de Sevilla, España
Fuente, Maria	Universidad de Valladolid, España
Galar, Mikel	Universidad Pública de Navarra, España
García-Lapresta, José Luis	Universidad de Valladolid, España
Godo, Lluis	IIIA - CSIC, España
Gomez, Daniel	Universidad Complutense de Madrid, España
Gonzalez, Antonio	Universidad de Granada, España
Indurain, Esteban	Universidad Pública de Navarra, España
Jurío, Aránzazu	Universidad Pública de Navarra, España
Lamata, Maria Teresa	Universidad de Granada, España
Liern, Vicente	Universidad de Valencia, España
Llamazares, Bonifacio	Universidad de Valladolid, España
Lopez-Molina, Carlos	Universidad Pública de Navarra., España
Magdalena, Luis	Universidad Politécnica de Madrid, España
Martínez, Luis	Universidad de Jaén, España
Marín, Nicolás	Universidad de Granada, España
Massanet, Sebastià	Universidad de las Islas Baleares, España
Medina, Jesús	Universidad de Cádiz, España
Molina, Jose M.	Universidad Carlos III de Madrid, España
Montero, Javier	Universidad Complutense de Madrid, España
Olivas, Jose Angel	Universidad de Castilla-La Mancha, España
Paternain, Daniel	Universidad Pública de Navrra, España

Perez, Raul	Universidad de Granada, España
Pomares, Hector	Universidad de Granada, España
Porcel, Carlos	Universidad de Granada, España
Pradera, Ana	Universidad Rey Juan Carlos, España
Pérez-Fernández, Raúl	Universidad de Oviedo, España
Ramírez Poussa, Eloisa	Universidad de Cádiz, España
Recasens, Jordi	Universidad Politécnica de Cataluña, España
Riera, Juan Vicente	Universidad de las Islas Baleares, España
Romero, Francisco P.	Universidad de Castilla-La Mancha, España
Sainz-Palmero, Gregorio	Universidad de Valladolid, España
Sanchez, Daniel	Universidad de Granada, España
Sanchez, Luciano	Universidad de Oviedo, España
Sanz Delgado, José Antonio	Universidad Pública de Navarra, España
Serrano-Guerrero, Jesus	Universidad de Castilla-La Mancha, España
Sicilia, Miguel-Angel	Universidad de Alcalá, España
Sobrino Cerdeiriña, Alejandro	Universidad de Santiago de Compostela, España
Sánchez Solano, Santiago	IMSE-CNM, España
Torra, Vicenc	Universidad de Skövde, Suecia
Torrens, Joan	Universidad de las Islas Baleares, España
Valls, Aida	Universidad Rovira i Virgili, España
Verdegay, José Luis	Universidad de Granada, España

MAEB

Alvarez, Ada	Universidad Autónoma de Nuevo León, México
Amaya, Jhon Edgar	Universidad Nacional Experimental del Táchira, Venezuela
Araujo, Lourdes	Universidad Nacional de Educación a Distancia, España
Bautista, Joaquín	Universitat Politècnica de Catalunya, España
Bello Orgaz, Gema	Universidad Politécnica de Madrid, España
Blum, Christian	Consejo Superior de Investigaciones Científicas, España
Brito, Julio	Universidad de La Laguna, España
Caballero, Rafael	Universidad de Málaga, España
Camacho, Carlos	Universidad Politécnica de Madrid, España
Casado, Silvia	Universidad de Burgos, España
Castillo, Pedro	Universidad de Granada, España
Chavez, Francisco	Universidad de Extremadura, España
Chica, Manuel	Universidad de Granada, España
Coello Coello, Carlos	CINVESTAV-IPN, México
Colmenar, José Manuel	Universidad Rey Juan Carlos, España
Colomine D., Feijoo	Universidad Nacional experimental del Táchira, Venezuela
Damas, Sergio	Universidad de Granada, España
Del-Jesus, María-José	Universidad de Jaén, España
Dorronsoro, Bernabe	Universidad de Cádiz, España
Duarte, Abraham	Universidad Rey Juan Carlos, España
Duro, Richard	Universidade da Coruna, España
Egea, Jose A.	CEBAS-CSIC, España
Faulin, Javier	Universidad Pública de Navarra, España
Fernandez De Vega, Francisco	Universidad de Extremadura, España
Fernández, Alberto	Universidad de Granada, España
Fernández Leiva, Antonio J.	Universidad de Málaga, España
Gámez, José Antonio	Universidad de Castilla-La Mancha, España
García, Pablo	Universidad de Granada, España
García Pardo, Eduardo	Universidad Politécnica de Madrid, España
García-Martínez, Carlos	Universidad de Córdoba, España
García-Pedrajas, Nicolás	Universidad de Córdoba, España
Gomez Romero, Juan	Universidad de Granada, España
Gonzalez, Antonio	Universidad de Granada, España
Gonzalez, Pedro	Universidad de Jaén, España
Gonzalez-Pardo, Antonio	Universidad Rey Juan Carlos, España
Gutierrez, Pedro Antonio	Universidad de Córdoba, España

Hernández-Díaz, Alfredo G.	Universidad Pablo de Olavide, España
Herrera, Francisco	Universidad de Granada, España
Hervas, Cesar	Universidad de Córdoba, España
Hidalgo, Ignacio	Universidad Complutense de Madrid, España
Huertas, Javier	Universidad Politécnica de Madrid, España
Juan, Angel	Universitat Oberta de Catalunya, España
Laguna, Manuel	Universidad de Colorado Boulder, Estados Unidos
Larranaga, Pedro	Universidad de Madrid, España
Luna, Francisco	Universidad de Málaga, España
Luna, Jose Maria	Universidad de Córdoba, España
Luque, Mariano	Universidad de Málaga, España
Luque-Baena, Rafael	Universidad de Extremadura, España
Magdalena, Luis	Universidad Politécnica de Madrid, España
Martín, Alejandro	Universidad Politécnica de Madrid, España
Melian, Belen	Universidad de La Laguna, España
Mendiburu, Alexander	Universidad del País Vasco, España
Mesejo, Pablo	Universidad de Granada, España
Molina-Solana, Miguel	Universidad de Granada, España
Moreno, José-A.	Universidad de La Laguna, España
Nebro, Antonio J.	Universidad de Málaga, España
Pacheco, Joaquín	Universidad de Burgos, España
Peiró, Juanjo	Universidad de Valencia, España
Pelta, David	Universidad de Granada, España
Peregrin, Antonio	Universidad de Huelva, España
Perez, Raul	Universidad de Granada, España
Ramalhinho, Helena	Universitat Pompeu Fabra, España
Ramírez-Atencia, Cristian	Universidad Politécnica de Madrid, España
Riquelme, José C.	Universidad de Sevilla, España
Risco-Martín, José L.	Universidad Complutense de Madrid, España
Rivas Santos, Victor Manuel	Universidad de Jaén, España
Rodríguez, David	Universidad Nacional Experimental del Táchira, Venezuela
Rodríguez-Fernández, Víctor	Universidad Politécnica de Madrid, España
Romano, Carlos-Andrés	Universidad Politécnica de Valencia, España
Rosete, Alejandro	Instituto Superior Politécnico "José Antonio Echevarría", Cuba
Rosso, Paolo	Universitat Politècnica de València, España
Ruiz, Ruben	Universitat Politècnica de València, España
Ríos-Mercado, Roger Z.	Universidad Autónoma de Nuevo León, México
Salcedo-Sanz, Sancho	Universidad de Alcalá, España
Sanchez, Luciano	Universidad de Oviedo, España
Santana, Roberto	Universidad del País Vasco, España
Stützle, Thomas	Université Libre de Bruxelles, Bélgica
Sáez, Yago	Universidad Carlos III de Madrid, España
Trujillo, Leonardo	Instituto Tecnológico de Tijuana, México
Udias, Angel	Universidad Rey Juan Carlos, España
Vega-Rodríguez, Miguel A.	Universidad de Extremadura, España
Ventura, Sebastián	Universidad de Córdoba, España
Verdegay, José Luis	Universidad de Granada, España
Villar, Pedro	Universidad de Granada, España
Villegas-Cortez, Juan	Universidad Autónoma Metropolitana, México
Winter, Gabriel	Universidad de Las Palmas de Gran Canaria, España
Zafra Gómez, Amelia	Universidad de Córdoba, España
Álvarez-Valdés, Ramón	Universidad de Valencia, España

TAMIDA

Arbelaitz, Olatz	Universidad del País Vasco, España
Bahamonde, Antonio	Universidad de Oviedo, España
Cano, Alberto	Virginia Commonwealth University, Estados Unidos
Corchado, Emilio	Universidad de Salamanca, España

Cordón, Óscar	Universidad de Granada, España
del Campo, José	Universidad de Málaga, España
Del Coz, Juan José	Universidad de Oviedo, España
del Jesús, Mª José	Universidad de Jaén, España
Ferri, Francesc J.	Universidad de Valencia, España
García López, Salvador	Universidad de Granada, España
García Pedrajas, Nicolás	Universidad de Córdoba, España
Gibert, Karina	Universidad Politécnica de Catalunya, España
González, Pedro	Universidad de Jaén, España
Hernández Orallo, José	Universidad Politécnica de Valencia, España
Herrera, Francisco	Universidad de Granada, España
Hervás, César	Universidad de Córdoba, España
Larrañaga, Pedro	Universidad Politécnica de Madrid, España
Lozano, José Antonio	Universidad del País Vasco, España
Martínez Ballesteros, María	Universidad de Sevilla, España
Martínez Álvarez, Francisco	Universidad Pablo de Olavide, España
Pérez-Ortiz, María	University College London, Reino Unido
Ramírez, Mª José	Universitat Politècnica de València, España
Rodríguez, Juan J.	Universidad de Burgos, España
Sànchez i Marrè, Miquel	Universidad Politécnica de Cataluña, España
Ventura, Sebastián	Universidad de Córdoba, España

Taller IABiomed

Araujo, Lourdes	Universidad Nacional de Educación a Distancia, España
Díez, Francisco Javier	Universidad Nacional de Educación a Distancia, España
Fernández-Breis, Jesualdo Tomás	Universidad de Murcia, España
Guijarro-Berdiñas, Bertha	Universidad de La Coruña, España
Larranaga, Pedro	Universidad de Madrid, España
López, Beatriz	Universidad de Girona, España
Martinez Tomas, Rafael	Universidad Nacional de Educación a Distancia, España
Muguerza, Javier	Universidad del País Vasco, España
Palma, Jose	Universidad de Murcia, España
Rodriguez, Alejandro	Universidad Politécnica de Madrid, España
Taboada, Maria	Universidad de Santiago de Compostela, España
Vellido, Alfredo	Universitat Politècnica de Catalunya, España
Villar, José Ramón	Universidad de Oviedo, España

Taller IA-SIT

Armingol Moreno, Jose Maria	Universidad Carlos III de Madrid, España
Molina Lopez, Jose Manuel	Universidad Carlos III de Madrid, España

Taller AI4D

Armuelles, Ivan	Universidad de Panama, Panama
Carbó, Javier	Universidad Carlos III de Madrid, España
Corrales, Juan Carlos	Universidad del Cauca, Colombia
Lopez-Vargas, Ascension	Universidad Carlos III de Madrid, España
Škrjanc, Igor	Universidad de Ljubljana, Eslovenia

Taller QUARTI

DAHI, Zakaria Abdelmoiz	Universidad de Málaga, España
Garcia de Andoin, Mikel	Universidad del País Vasco, España
Torrontegui, Erik	Universidad Carlos III de Madrid, España

Taller AGM

Bazazian, Dena	Universidad de Bristol, Reino Unido
Keane, Mark	University College Dublin, Irlanda
Pintore, Gianni	Center for Advanced Studies, Research and Development in Sardinia, Italia

Ribelles, Jose	Universitat Jaume I, España
Seoane, Antonio	Universidade da Coruña, España

Taller SCAI

Cintrano, Christian	Universidad de Málaga, España
Morell, José Angel	Universidad de Málaga, España
Muñoz, Antonio	Universidad de Málaga, España
Rossit, Diego Gabriel	Universidad Nacional del Sur - CONICET, Argentina

Jurado de Competiciones

Corchado, Juan Manuel	Universidad de Salamanca, España
Cordón, Óscar	Universidad de Granada, España
del Jesús, María José	Universidad de Jaén, España
Guijarro-Berdiñas, Bertha	Universidad de La Coruña, España
Luaces, Óscar	Universidad de Oviedo, España
Onaindia, Eva	Universitat Politècnica de València, España
Troncoso, Alicia	Universidad Pablo de Olavide, España

Jurado de Doctoral Consortium

Duarte, Abraham	Universidad Rey Juan Carlos, España
Bahamonde, Antonio	Universidad Oviedo, España
Hervás, César	Universidad Córdoba, España
Bustince, Humberto	Universidad Pública de Navarra, España
Gámez, José Antonio	Universidad Castilla-La Mancha, España
Lozano, José Antonio	Universidad del País Vasco, España
Riquelme, José	Universidad Sevilla, España
Rodríguez, Juan José	Universidad de Burgos, España
Gibert, Karina	Universidad Politécnica de Catalunya, España
Magdalena, Luis	Universidad Politécnica de Madrid, España
del Jesús, María José	Universidad Jaén, España
Ramírez, María José	Universidad Politécnica de Valencia, España
Cordón, Óscar	Universidad de Granada, España
Larrañaga, Pedro	Universidad Politécnica de Madrid, España
Meseguer, Pedro	IIIA-CSIC, España
Martí, Rafael	Universidad Valencia, España
Bolón, Verónica	Universidad de La Coruña, España

Índice general

Presentación de CAEPIA 20/21 III

Presentación de la Presidenta de AEPIA V

Ponentes Plenarios VII

Organización IX

Comités de Programa XIII

XIX Conferencia de la Asociación Española para la Inteligencia Artificial

Sesión General

Multi-Objective Bayesian Optimization approach for Environmental Monitoring using an Autonomous Surface Vehicle: Ypacarai Lake Case Study 5
Federico Peralta, Samuel Yanes, Daniel Gutierrez Reina, Sergio Toral

Genetic Enhanced Model-based Deep Reinforcement Learning for Informative Path Planning in the Lake Ypacaraí Scenario 11
Samuel Yanes, Federico Peralta, Daniel Gutiérrez Reina, Sergio Toral

Bi-objective Evolutionary Approach for WaterResource Patrolling Problem with Autonomous 17
Samuel Yanes, Federico Peralta, Daniel Gutiérrez, Alejandro Tapia, Álvaro Rodriguez, Sergio Toral

Sistema Inteligente para la Gestión Eficiente de Recogida de Residuos Plásticos 23
Juan José Guerrero, Javier Ferrer, Rubén Saborido, Enrique Alba

Sistema de Recomendación con Explicaciones Basadas en Texto 29
Pablo Pérez-Núñez, Antonio Bahamonde, Oscar Luaces, Jorge Díez

Método XAI basado en agrupamiento para la explicación de errores en clasificación 35
Aurora Ramírez, Sebastián Ventura, José Raúl Romero

Revisión y análisis de conceptos en Inteligencia Artificial Explicable. Una aproximación a la unificación de la terminología 41
Ángela Escobar Mimbrera, Pedro González García, María José Del Jesús Díaz

Challenges of Definition Extraction in Spanish Legal Texts 47
Karen Leticia Vázquez-Flores, Patricia Martín Chozas, Elena Montiel Ponsola

Bilingual Dataset for Information Retrieval and Question Answering over the Spanish Workers Statute 53
Pablo Calleja, Patricia Martín Chozas, Elena Montiel-Ponsoda, Victor Rodriguez Doncel

Towards Document Entity Recognition using Close Domain Transfer Learning 58
Adrián Inés Armas, César Domínguez, Jónathan Heras, Eloy Javier Mata, Vico Pascual

Runners detection based on trajectory analysis 64
Jonay Sarmiento-Ramírez, Javier Lorenzo-Navarro, Modesto Castrillon-Santana, Elena Sánchez-Nielsen

Evaluación Estética de Fotografías vía Deep Learning y enfoques probabilísticos 70
Luis González Naharro, José Miguel Puerta Callejón, Fernando Rubio Perona, Maria Julia Flores Gallego

Prediction of Epiretinal Membrane from Retinal Fundus Images Using Deep Learning 76
Ángela Casado-García, Manuel García-Domínguez, Jónathan Heras, Adrián Inés, Didac Royo, Miguel Ángel Zapata

Evaluation of a CNN+LSTM system for theclassification of hand-washing steps 77
Kevin Cikel, Mario Arzamendia, Derlis Gregor, Daniel Gutiérrez, Sergio Toral

Ultramétricas para la comparación contextualizada de imágenes binarias 83
Sara Iglesias-Rey, Felipe Antunes-Santos, Arnau Mir-Fuentes, Carlos Lopez-Molina

Towards Combining Object Detection and Text Classification Models for Form Entity Recognition 89
María Villota, Gonzalo Santamaría, César Domínguez, Jónathan Heras, Eloy Mata, Vico Pascual

AutoML para la composición de workflows diversos mediante programación genética gramatical 90
Rafael Barbudo Lunar, Sebastián Ventura, José Raúl Romero

Autonomous Learning: Tackling Expansive Mutating Domains 96
Harshal Bharatia

Learning A Swarm Formation Policy using Deep Reinforcement Learning 102
Jasmina Rais, Fidel Aznar

Generalized SMOTE: A universal generation oversampling technique for all data types in imbalanced learning 108
Carlos Cernuda, Daniel Reguera-Bakhache, Aitor Aguirre, Mikel Iturbe, Iñaki Garitano, Urko Zurutuza

Deep Reinforcement and Imitation Learning for Self-Driving Tasks 114
Sergio Hernández-García, Alfredo Cuesta-Infante

A Proposal to Integrate Deep Q-Learning with Automated Planning to Improve the Performance of a Planning-based Agent 115
Carlos Núñez-Molina, Ignacio Vellido, Vladislav Nikolov-Vasilev, Raúl Pérez, Juan Fdez-Olivares

Ontologies for run-time self-adaptation of mobile robotic systems 116
Esther Aguado, Ricardo Sanz, Claudio Rossi

Analysing Ontological Requirements: A Journey from Requirements to Code and Back 122
Alba Fernandez-Izquierdo, María Poveda-Villalón, Raúl García-Castro

Modelos gráficos probabilísticos basados en mixturas para el problema de partial label ranking 128
Juan C. Alfaro, Juan A. Aledo, Jose A. Gámez

LabelDetection: Simplifying the Use and Construction of Deep Detection Models 134
Ángela Casado-García, Jónathan Heras

Algoritmos para el problema del bandido multi-objetivo basados en la escalarización de Chebyshev 135
Lawrence Mandow, Sergio Martín-Albo, Jose-Luis Perez De La Cruz

Caracterización de conjuntos de puntos mediante cierres no convexos subdivisibles 141
David Novoa-Paradela, Oscar Fontenla-Romero, Bertha Guijarro-Berdiñas

A memetic algorithm to minimize the total weighted tardiness in the fuzzy flexible job shop 147
Pablo García Gómez, María Camino Rodríguez Vela, Inés González Rodríguez

Estimación de las tensiones residuales a nivel de grano en una muestra cilíndrica templada de aleación de aluminio AA5083 mediante programación genética 153
Laura Millán, Gabriel Kronberger, Ricardo Fernández Serrano, Oscar Garnica, Patricie Halodova, Gaspar González Doncel, Ignacio Hidalgo

Towards Automatic Bayesian Optimization: A First Step Involving Acquisition Functions 159
Luis C. Jariego Pérez, Eduardo C. Garrido-Merchán

Diseñando múltiples genes para una proteína mediante el algoritmo memético SFLA 160
Belen Gonzalez-Sanchez, Miguel A. Vega-Rodríguez, Sergio Santander-Jiménez

Recognition of Teaching Activities from University Lecture Transcriptions 166
Daniel Diosdado, Alberto Romero, Eva Onaindia

Human Activity Recognition with Capsule Networks 167
Laura Llopis-Ibor, Alfredo Cuesta-Infante, César Beltrán-Royo, Juan José Pantrigo

Identification of Contact Failures in Multilayered Media via Inverse Problem and Artificial Neural Networks 168
Lucas Correia da Silva Jardim, Diego Campos Knupp, Roberto Pinheiro Domingos, Luiz Alberto da Silva Abreu, Carlos Cruz Corona, Antônio José da Silva Neto

Estimación de la biomasa en acuicultura utilizando redes neuronales convolucionales 174
Samuel Lopez, Antonio Peregrin, Juan Carlos Gutierrez, Inmaculada Pulido, Jairo Castro, Fernando Gomez, Alejandro Garrocho

An unsupervised anomaly detection strategy for industrial processes under data shift scenarios 180
Xabier Barrenetxea Berasategi, Alberto Diez-Olivan, Óscar Cordón

Train Route Planning as a Multi-Agent Path Finding Problem 186
Mauricio Salerno, Yolanda E-Martín, Raquel Fuentetaja, Alba Gragera, Alberto Pozanco, Daniel Borrajo

Asynchronous Vector Iteration in Multi-Objective Markov Decision Processes 187
Ekaterina Sedova, Lawrence Mandow, José-Luis Pérez-de-la-Cruz

A Similarity Measure of Gaussian Process Predictive Distributions 188
Lucía Asencio Martín, Eduardo C. Garrido-Merchán

Towards Fairness in Classification: Comparison of Methods to Decrease Bias 189
Maitane Martínez-Eguiluz, Oier Irazabal-Urrutia, Olatz Arbelaitz-Gallego

Modeling Administrative Discretion Using Goal-Directed Answer Set Programming 190
Joaquín Arias, Mar Moreno-Rebato, José A. Rodríguez-García, Sascha Ossowski

Automatic Generation of Interrelated Organisms on Virtual Environments 191
Santiago Pacheco, Nicolás Ottonello, Sergio Nesmachnow

An analysis of the indexes measuring the agreement of a profile of rankings 192
Noelia Rico, Camino R. Vela, Irene Diaz

Trabajos Destacados

A Hierarchical Architecture for Recognising Intentionality in Mental Tasks on a Brain-Computer Interface . 201
Asier Salazar-Ramirez, Jose I. Martin, Raquel Martinez Rodriguez, Andoni Arruti, Javier Muguerza, Basilio Sierra

A Self-Paced Relaxation Response Detection System Based on Galvanic Skin Response Analysis 203
Raquel Martinez Rodriguez, Asier Salazar, Jose Ignacio Martin, Andoni Arruti, Eloy Irigoyen, Javier Muguerza

Improving Bayesian Inference Efficiency for Sensory Anomaly Detection and Recovery in Mobile Robots . . . 205
Manuel Castellano-Quero, Juan-Antonio Fernández-Madrigal, Alfonso García-Cerezo

Template-free detection and classification of membrane-bound complexes in cryo-electron tomograms 207
Antonio Martinez-Sanchez

Improving fake faces detection toprevent fraudulent registrations 209
Luis Carabe, Eduardo Cermeño

Hair Segmentation and Removal in Dermoscopic Images using Deep Learning 211
Lidia Talavera-Martínez, Pedro Bibiloni, Manuel González-Hidalgo

Navigating at the microscopic scale with Reinforcement Learning 213
Santiago Muiños-Landín, Alexander Fischer, Viktor Holubek, Frank Cichos

Online Learning for Probabilistic Load Forecasting 215
Verónica Álvarez, Santiago Mazuelas, José A. Lozano

CCE: An ensemble architecture based on coupled ANN for solving multiclass problems 217
M. Paz Sesmero, Juan M. Alonso-Weber, Araceli Sanchis

Reliable and Fast Recurrent Neural NetworkArchitecture Optimization 219
Andrés Camero, Jamal Toutouh, Enrique Alba

An AI-Powered System for Residential Demand Response 221
Iker Esnaola-Gonzalez, Marko Jelić, Dea Pujić, Francisco Javier Diez, Nikola Tomasevic

Sobre órdenes admisibles en el conjunto de números borrosos discretos y su aplicación en problemas de toma de decisiones 223
Juan Vicente Riera, Sebastia Massanet, Humberto Bustince, Francisco Javier Fernandez

Aprendizaje Federado y Privacidad Diferencial: Análisis de herramientas software, el framework Sherpa.ai FL y guías metodológicas para preservar la privacidad de datos 225
Nuria Rodríguez-Barroso, Eugenio Martínez-Cámara, María V. Luzón, Francisco Herrera

FEPDS: Una propuesta para la extracción de patrones emergentes difusos en flujos continuos de datos 227
Angel M. Garcia-Vico, Huseyin Seker, Cristobal J. Carmona, Pedro Gonzalez, María José Del Jesús

Miniaturización de aplicaciones JavaScript para el Internet de las Cosas 229
Rubén Saborido

E2PAMEA: un algoritmo evolutivo para la extracción eficiente de patrones emergentes difusos en entornos big data 231
Angel M. Garcia-Vico, David Elizondo, Francisco Charte, Pedro Gonzalez, Cristobal J. Carmona

Fostering Diversity in Spatial Evolutionary Generative Adversarial Networks 233
Jamal Toutouh, Erik Hemberg, Una-May O'Reilly

Revisiting data complexity metrics based on morphology for overlap and imbalance: snapshot, new Overlap Number of Balls metrics and singular problems prospect 235
José Daniel Pascual Triana, David Charte, Marta Andrés Arroyo, Alberto Fernández, Francisco Herrera

XX Congreso Español sobre Tecnologías y Lógica Fuzzy

Sesión General

Multiple fuzzy Sugeno λ-measures in networks 241
Inmaculada Gutiérrez, Daniel Gomez, Javier Castro, Rosa Espínola

Explicaciones factuales y contrafactuales en árboles de clasificación difusos 247
Guillermo Fernández, Juan Aledo, José Gámez, José Puerta

Comparando variabilidades de conjuntos aleatorios 253
Juan Jesús Salamanca, Susana Montes

A fuzzy probability logic for compound conditionals 256
Tommaso Flaminio, Lluis Godo

Empleo de contextos L-fuzzy en el diseño de materiales de origen renovable 262
Itsaso Leceta, Cristina Alcalde, Marta Urdanpilleta, Pedro Guerrero, Koro de la Caba, Ana Burusco

Una versión ternaria del algoritmo de Quine McCluskey para la minimización de base de reglas difusas 268
Leonardo Jara, Antonio González, Raúl Pérez

Prometheus: Harnessing Fuzzy Logic and Natural Language for Human-centric Explainable Artificial Intelligence 274
Ettore Mariotti, Jose M. Alonso-Moral, Albert Gatt

Modelando el comportamiento del consumidor con el modelo de 2-tuplas lingüísticas y una heurística basada en el conocimiento de marca 280
Jesús Giráldez-Cru, Manuel Chica, Óscar Cordón, Francisco Herrera

Generación de descripciones lingüísticas de la contaminación acústica diaria en zonas urbanas 286
Juan Moreno Garcia, Luis Jimenez Linares, Luis Rodriguez-Benitez

Un sistema para la descripción automática en lenguaje natural de gráficas de sectores: aplicación en datos de calidad del aire 292
Andrea Cascallar-Fuentes, Javier Gallego-Fernández, Alejandro Ramos-Soto, Anthony Saunders, Alberto Bugarín-Diz

Evaluación empírica de modelos de cuantificación borrosa aplicados a un agente conversacional 298
Mariña Canabal-Juanatey, Jose M. Alonso-Moral, Alejandro Catala, Alberto Bugarín-Diz

Hypothesis Scoring and Model Refinement Strategies for FM-based RANSAC 304
Alberto Ortiz, Esaú Ortiz, Juan José Miñana, Óscar Valero

Análisis de distintos tipos de tendencias en sucesiones de contextos L-fuzzy 305
Cristina Alcalde, Ana Burusco

On the standard fuzzy metric: generalizations and application to model estimation 311
Juan-José Miñana, Alberto Ortiz, Esaú Ortiz, Oscar Valero

Uso de t-normas para el estudio de la convexidad en conjuntos difusos intervalo-valuados 317
Pedro Huidobro, Pedro Alonso, Humberto Bustince, Vladimír Janis, Susana Montes

Sesión especial: Toma de decisiones con información difusa

Preferencias no lineales en Toma de Decisión en Grupo. Amplificación de Valores Extremos 323
Diego García-Zamora, Álvaro Labella Romero, Rosa M. Rodríguez, Luis Martínez

Estructuras de preferencia para relaciones recíprocas borrosas: caracterización y compatibilidad 329
Fabian Alberto Castiblanco Ruiz, Camilo Franco De Los Ríos, J. Tinguaro Rodríguez, Javier Montero

Generalización de intervalos en matrices de riesgo imprecisas 331
María Martínez, Juan Baz, Raul Perez-Fernandez, Susana Montes

Consenso en Entornos Académicos Virtuales: Toma de Decisiones Grupales en el Trabajo de Clase Online .. 337
Aitor Manuel Orellana, Antonio Velez-Estevez, Manuel Jesus Cobo, Ignacio Javier Perez

Sesión especial: Funciones de agregación y preagregación y sus aplicaciones

Mapas de características adicionales en redes neuronales profundas a partir de agregaciones basadas en orden 345
Iris Dominguez-Catena, Daniel Paternain, Mikel Galar

Learning OWA weights by combining fuzzy quantifiers with empirical data 351
Álvaro Cristóbal, Ignacio Huitzil, Fernando Bobillo

Integrales de Choquet y capacidades 2-aditivas en la construcción de indicadores sintéticos 357
Patrizia Pérez-Asurmendi, Bonifacio Llamazares Rodríguez

Extensión multidimensional de la integral de Choquet discreta y su aplicación en redes neuronales recurrentes 358
Mikel Ferrero-Jaurrieta, Iosu Rodríguez-Martínez, Graçaliz Pereira Dimuro, Javier Fernández, Humberto Bustince

Monotonicity in Non-deterministic Computable Aggregations 364
Luis Magdalena, Daniel Gomez, Luis Garmendia, Javier Montero

Optimizando Desviaciones Moderadas Ponderadas para Interfaces Cerebro Ordenador 370
Javier Fumanal Idocin, Carmen Vidaurre, Marisol Gómez, Asier Urio, Graçaliz Pereira-Dimuro, Humberto Bustince

Operador de comparación de elementos multivaluados basado en Funciones de Equivalencia Restringida 376
Aitor Castillo-López, Carlos Lopez-Molina, Javier Fernandez, Humberto Bustince, Mikel Sesma-Sara

Trabajos Destacados

Computing with Comparative Linguistic Expressions and Symbolic Translation for Decision Making: ELICIT Information . . . 385
Álvaro Labella Romero, Rosa M. Rodríguez, Luis Martínez

Aggregation of fuzzy quasi-metrics . . . 387
Tatiana Pedraza, Jesús Rodríguez-López, Óscar Valero

A graded notion of quasi-intents . . . 389
Manuel Ojeda-Hernandez, Inma P. Cabrera, Pablo Cordero

Community Detection Problem Based on Polarization Measures: An Application to Twitter. The COVID-19 Case in Spain. A Review . . . 391
Inmaculada Gutiérrez, Juan Antonio Guevara, Daniel Gómez, Javier Castro, Rosa Espínola

Rough-Set-Driven Approach for Attribute Reduction in Fuzzy Formal Concept Analysis . . . 393
María José Benítez-Caballero, Jesús Medina, Eloisa Ramírez-Poussa, Dominik Slezak

Measuring Consistency of Fuzzy Logic Theories . . . 395
Nicolas Madrid, Manuel Ojeda-Aciego

Detecting equivalence classes being sublattices from an attribute reduction in FCA . . . 397
Roberto García-Aragón, Jesús Medina, Eloisa Ramírez Poussa

Non-monotonic multi-adjoint logic programming with constraints . . . 398
M. Eugenia Cornejo, David Lobo, Jesús Medina

XIV Congreso Español de Metaheurísticas, Algoritmos Evolutivos y Bioinspirados

Sesión General

On the role of the gradient optimizer on evolved GANs . . . 405
Unai Garciarena, Alexander Mendiburu, Roberto Santana

Aplicación de técnicas de aprendizaje profundo al reconocimiento óptico de partituras SATB . . . 411
Martín Morita Hernández, Francisco Fernández de Vega, Juan Villegas Cortez

Optimización Evolutiva de parámetros sobre Random Forest en Entornos de Big Data . . . 417
Miguel Angel Rodriguez, Antonio Peregrin

Un estudio de los diferentes criterios aplicables al problema de resumen extractivo multi-documento . . . 423
Jesus M. Sanchez-Gomez, Miguel A. Vega-Rodríguez, Carlos J. Pérez

Algoritmos Evolutivos con Memoria Elite para la Selección de Carteras de Inversión . . . 429
Feijoo Colomine, Carlos Cotta, Antonio J. Fernández Leiva

Un Enfoque Evolutivo Multi-Operador para el Diseño de Rutas Turísticas con Nivel de Interés Dependiente del Tiempo . . . 434
David Pelta, Juan Carlos Ortegón, Marcelino Cabrera

Cooperación Inter-algorítmica Multiobjetivo para Reconstrucción Filogenética . . . 440
Sergio Santander-Jiménez, Miguel A. Vega-Rodríguez, Leonel Sousa

El efecto de la multiconectividad en el problema del apagado selectivo de redes 5G ultradensas . . . 446
Alejandro Quiñones, Jesús Galeano, Pablo Zapata, Francisco Luna, Javier Carmona, Juan Francisco Valenzuela

Influence of the Alternative Objective Functions in the Optimization of the Cyclic Cutwidth Minimization Problem . . . 452
Sergio Cavero, Eduardo G. Pardo, Abraham Duarte

Bayesian Optimization for Permutations Spaces: A preliminary approach . . . 453
Daniel Jiménez, Josu Ceberio, Jose Antonio Lozano

Analyzing Elementary Landscapes in Graph Partitioning Problems 459
Ignacio Dorado Llamas, Josu Ceberio Uribe

A Matheuristic Approach to the Interval Job Shop Scheduling Problem 465
Sezin Afsar, Juan Jose Palacios, Camino R. Vela

Solving the Permutation Heijunka Flow Shop Scheduling Problem with Non-unit Demands for Jobs 471
Joaquín Bautista-Valhondo

Combinando GRASP con Iterated Greedy para el problema de la máxima intersección de k-conjuntos 472
Alejandra Casado Ceballos, Sergio Pérez Peló, Jesús Sánchez-Oro Calvo, Abraham Duarte Muñoz

Optimización multiobjetivo basada en el algoritmo ABC para la codificación de proteínas 478
Belen Gonzalez-Sanchez, Miguel A. Vega-Rodríguez, Sergio Santander-Jiménez

Impacto Ambiental de la Experimentación con Metaheurísticas: Un Análisis Coste-Beneficio 484
Pavel Novoa-Hernández, David A. Pelta

Un enfoque multi-objetivo para el problema de la máxima diversidad 490
Pedro Casas-Martínez, Alejandra Casado-Ceballos, Jesús Sánchez-Oro, Eduardo G. Pardo

Sesión especial: Algoritmos Bioinspirados Profundos

Una Visión Panorámica de la Meta-Cooperación Profunda con Algoritmos Meméticos 499
Jhon Edgar Amaya, Carlos Cotta, Antonio J. Fernández Leiva, Pablo García Sánchez

En busca de algoritmos genéticos eficientes desde el punto de vista energético 505
Jorge Alvarado Díaz, Francisco Chávez, Pedro Castillo, Josefa Díaz, Francisco Fernández de Vega

Clusterig Cosmológico: un enfoque del clustering gravitacional clásico inspidado en la estructura y dinámica del cosmos a gran escala 511
Aitor Castillo-López, Javier Fumanal-Idocin, Javier Fernandez, Humberto Bustince

Cartografía de un Subcampo Científico: Un Estudio sobre la Optimización de Carteras de Inversión 517
Feijoo Colomine, Carlos Cotta, Antonio J. Fernández Leiva

A la desinformación le gusta la compañía: Representación de bulos de Twitter sobre la COVID-19 mediante embeddings 523
Guillermo Villar-Rodríguez, Javier Huertas-Tato, Alejandro Martín, David Camacho

Mejorando Redes Profundas para Aprendizaje por Transferencia mediante Capas Dispersas optimizadas con Algoritmos Evolutivos 529
Javier Poyatos Amador, Daniel Molina Cabrera, Aritz Martínez, Siham Tabik, Javier Del Ser Lorente, Francisco Herrera Triguero

Sesión especial: Aplicaciones en Medicina y Biotecnología

Predicción de la glucosa en sangre mediante un sistema basado en reglas difusas TSK de dos fases 537
Jorge Alvarado Díaz, José Manuel Velasco, Francisco Chávez, Almudena Sánchez, Francisco Fernández de Vega

Modelo híbrido basado en aprendizaje profundo de creación y etiquetado de células escamosas procedentes de citologías cervicovaginales 543
Andrés Bueno-Crespo, Ana Ortiz-González, Raquel Martínez España, José Martínez-Más, Miguel Ángel Gragera, Manuel Remezal-Solano, Juan-Pedro Martínez-Cendán, Sebastián Ortiz-Reina, Andres Muñoz

Selección óptima de variables mediante computación evolutiva para algoritmos de clasificación 549
Daniel Parra, Alberto Gutiérrez, Oscar Garnica, Juan Lanchares, José Ignacio Hidalgo

Calibración y predicción probabilística de un modelo de crecimiento del cáncer de vejiga teniendo en cuenta la incertidumbre de los datos 555
Juan Carlos Cortés, Elena López-Navarro, Ana Moscardó-García, Rafael-J. Villanueva

Heurística para el desarrollo de un sistema experto epidemiológico aplicado al COVID-19 560
Beatriz González-Pérez, Concepción Nuñez, José L. Sánchez, Gabriel Valverde, Jose Manuel Velasco

Calibración de un modelo basado en agentes que describe la evolución de la resistencia del Acinetobacter baumannii a la colistina en la Ciudad de Valencia 566
Juan A. Aledo, Carlos Andreu-Vilarroig, Juan Carlos Cortés, Juan C. Orengo, Rafael-J. Villanueva

A Blood Glucose classification and prediction system by Neural Networks 572
Alvaro Delgado, Alejandro Varela, Félix Tena, Jose Manuel Velasco, J. Ignacio Hidalgo

Sesión especial: Estrategias de implementación eficiente de heurísticas y metaheurística

Heurísticas para la mejora de la mantenibilidad de proyectos software 581
Javier Yuste, Eduardo G. Pardo, Abraham Duarte

Un algoritmo eficiente para el problema de disposición de instalaciones en dos filas 587
Raúl Martín-Santamaría, Alberto Herrán, Abraham Duarte, José Manuel Colmenar

Algoritmos de estimación del tiempo de ventana enla recogida de pedidos en almacenes logísticos 593
Sergio Gil Borrás, Eduardo García Pardo, Antonio Alonso Ayuso, Abraham Duarte, Ernesto Jiménez Merino

Sesión especial: Metaheurísticas Aplicadas al Análisis de Redes Sociales

Nuevos algoritmos metaheurísticos para el análisis de la influencia de los usuarios en las redes sociales 601
Isaac Lozano-Osorio, Jesús Sánchez-Oro, Abraham Duarte, Óscar Cordón

Detección de comunidades con un enfoque multi-objetivo utilizando VNS 607
Sergio Pérez-Peló, Jesús Sánchez-Oro, Antonio Gonzalez-Pardo, Abraham Duarte

Análisis de Redes Sociales basado en las Conquistas de César Borgia 613
Javier Idocin, Oscar Cordon, Amparo Alonso-Betanzos, Humberto Bustince, Javier Fernandez

Evaluación de modelos multilinguales pre-entrenados en similitud semántica para la lucha contra la desinformación 619
Álvaro Huertas-García, Alejandro Martín, Javier Huertas-Tato, David Camacho

Trabajos Destacados

Hybrid GRASP-SO-PR algorithm. The closest vs. the most balanced 627
Ana Dolores López-Sánchez, Jesús Sánchez-Oro, Anna Martínez-Gavara, Alfredo G. Hernández-Díaz, Abraham Duarte

Complete Shipment constraints in Container Loading Problem 629
Iván Giménez-Palacios, Maria Teresa Alonso, Ramon Alvarez-Valdes, Francisco Parreño

Scatter search for the capacitated dispersion problem 631
Anna Martínez-Gavara, Jesús Sánchez-Oro, Rafael Marti

A review and empirical analysis of diversity and dispersion problems 633
Rafael Martí, Anna Martínez-Gavara, Sergio Pérez-Peló, Jesús Sánchez-Oro

X Simposio de Teoría y Aplicaciones de Minería de Datos

Sesión General

Nearest Neighbors-based Forecasting for Electricity Demand Time Series in Streaming 639
L. Melgar-García, D. Gutiérrez-Avilés, C. Rubio-Escudero, A. Troncoso

STree: a Single Multi-class Oblique Decision Tree Based on Support Vector Machines 640
Ricardo Montañana, José A. Gámez, José M. Puerta

Studying the Effect of Different L_p Norms in the Context of Time Series Ordinal Classification . . . 641
David Guijo-Rubio, Víctor Manuel Vargas, Pedro Antonio Gutiérrez, César Hervás-Martínez

Universal Adversarial Examples in Speech Command Classification . . . 642
Jon Vadillo, Roberto Santana

Automatic Labelling using QA . . . 648
Eugenia Taillefer, Manuel Baena-García, Rafael Morales-Bueno

Reemplazo de la función de pooling de Redes Neuronales Convolucionales por combinaciones lineales de funciones crecientes . . . 652
Iosu Rodríguez, Julio Lafuente, Mikel Sesma, Francisco Herrera, Pablo Ursúa, Humberto Bustince

Evaluation of the Transformer Architecture for Univariate Time Series Forecasting . . . 658
Pedro Lara-Benítez, Luis Gallego-Ledesma, Manuel Carranza-García, José M. Luna-Romera

Electricity Consumption Time Series Forecasting Using Temporal Convolutional Networks . . . 659
J. F. Torres, M. J. Jiménez-Navarro, F. Martínez-Álvarez, A. Troncoso

ReLU-based Activations: Analysis and Experimental Study for Deep Learning . . . 660
Víctor Manuel Vargas, David Guijo-Rubio, Pedro Antonio Gutiérrez, César Hervás-Martínez

Clustering: Un paquete R para facilitar el análisis de algoritmos de agrupamiento . . . 661
Luis Alfonso Pérez Martos, Pedro González García, Cristóbal José Carmona del Jesús

Computer Science doctoral research in Spain. A relational perspective . . . 667
Adrián Arnaiz-Rodríguez, José Miguel Ramírez-Sanz, José Luis Garrido-Labrador, Alicia Olivares-Gil

Estudio comparativo de medidas de disimilitud para Clustering Multi-Instancia . . . 673
Aurora Esteban Toscano, Amelia Zafra Gómez, Sebastián Ventura Soto

A cluster merge algorithm to classify on/offline records of researchers in programming training . . . 679
Sonia Estévez, Victoria Lopez

Valoración de Anomalías Multi-paso Basada en Histogramas para Detección de Anomalías no Supervisada . 685
Ignacio Aguilera-Martos, Marta García-Barzana, Diego García-Gil, Jacinto Carrasco, David López, Manuel Arias-Rodil, Julián Luengo, Francisco Herrera

Estudio experimental sobre XGBoost, Rotation Forest y su combinación . . . 691
Francisco Javier González-Moya, Álvar Arnaiz-González, Juan J. Rodríguez

Estudio de estrategias basadas en vecinos para clasificación multi-instancia multi-etiqueta . . . 697
Eva Gibaja, Amelia Zafra

Comparing BERT against traditional machine learning text classification . . . 703
Santiago González-Carvajal, Eduardo C. Garrido-Merchán

Aprendizaje de clasificadores binarios justos mediante Optimización Multiobjetivo . . . 706
Jorge Casillas, David Villar

Análisis predictivo efectivo en centros de datos: una solución basada en inteligencia artificial . . . 712
David Garcia-Retuerta, Roberto Casado-Vara, Juan M. Corchado

Uso de técnicas de Inteligencia Artificial en el ámbito de la Atención Temprana . . . 717
Ignacio Sierra, Norberto Díaz-Díaz, Carlos D. Barranco, Rocio Carrasco-Villalón

Un sistema de apoyo al experto en actividad física basado en Metaheurísticas . . . 723
Joaquín Roiz Pagador, Roberto Ruiz, África Calvo-Lluch, Adán González-Reina

StreetQR: Informative assistance device for street name plates and places of interest . . . 729
Jesús Benito-Picazo, Jorge García-González, Gonzalo Ramos-Jiménez, Ezequiel López-Rubio

Personalización de Recomendaciones en TripAdvisor Usando Deep Learning . . . 735
Iñigo Luis López-Riobóo Botana, Amparo Alonso Betanzos, Verónica Bolón Canedo, Antonio Bahamonde Rionda

Sistema de Recomendación Transparente y Escrutable para Recomendaciones Personalizadas 741
Antonio Rodríguez-Revuelta, Pablo Pérez-Núñez, Jorge Díez

Automatic Head Appearance Recognition for Individuals Affected by PIMD 747
Carmen Campomanes-Álvarez, B. Rosario Campomanes-Álvarez, Pelayo Quirós

Selección de características en una base de datos sobre el cáncer colorrectal 753
José Antonio Delgado-Osuna, Daniel Ranchal-Parrado, Carlos García-Martínez, Sebastián Ventura

Identification of outliers through classification based on evolution indices and their application to COVID-19 757
Yury Andrea Jiménez Agudelo, Victoria Lopez, Walter Chanava

Open Data and Experimental data source combination's impact on underground water analysis in Isla Margarita, Venezuela 763
Karina Gibert, Dante Conti, Iñigo Arregui Pérez de Loza

Análisis descriptivo de cáncer de mama usando minería de datos 769
Antonio Manuel Trasierras, Jose Maria Luna, Sebastián Ventura

Aplicación de técnicas de Aprendizaje Automático para el análisis de datos de acciones de juego en voleibol 775
Francisco Aragón Royón, Elia Mercado Palomino, Aurelio Ureña Espa, José M. Benítez

Trabajos Destacados

Rotation Forest for Big Data 783
Mario Juez-Gil, Álvar Arnaiz-González, Juan J. Rodríguez, Carlos López-Nozal, César García-Osorio

Rotation Forest for Multi-target Regression 785
Juan J. Rodríguez, Mario Juez-Gil, Carlos López-Nozal, Álvar Arnaiz-González

Experimental evaluation of ensemble classifiers for imbalance in Big Data 787
Mario Juez-Gil, Álvar Arnaiz-González, Juan J. Rodríguez, César García-Osorio

Talleres, Competiciones y Doctoral Consortium

II Taller de Grupos de investigación españoles de IA en Biomedicina (IABiomed)

Methods and measures to quantify ICU patient heterogeneity 793
David Cuadrado, David Riaño, Josep Gomez, Alejandro Rodríguez, Maria Bodi

Neural Style Transfer to deal with the Domain Shift Problem on Glioblastoma Segmentation 795
Manuel García, Cesar Dominguez, Jónathan Heras, Eloy Javier Mata, Vico Pascual

Fall detection for healthy and autonomous elderly people 800
Mirko Fáñez Kertelj, José Ramón Villar, Enrique de La Cal, Víctor Gonzalez, Javier Sedano, Samad Barri Khojasteh

A platform for the design and execution of clinical data transformation and reasoning workflows 802
José Alberto Maldonado, Mar Marcos, Jesualdo Tomás Fernández-Breis, Vicente Miguel Giménez-Solano, María del Carmen Legaz-García, Begoña Martínez-Salvador

DISNET: extracting public phenotypical knowledge of diseases 804
Lucía Prieto Santamaría, Ernestina Menasalvas Ruiz, Alejandro Rodríguez-González

Using Machine Learning for the prediction of mutations in the initiation codon 806
Javier Castell Díaz, Francisco Abad-Navarro, Jesualdo T. Fernández-Breis, M. Eugenia de la Morena Barrio, Javier Corral de la Calle

A Helping Hand to CDSS Rule Technologies in Fighting Antibiotic Resistance for Hospital Settings 812
Bernardo Canovas-Segura, Natalia Iglesias, Antonio Morales, Jose M. Juarez, Manuel Campos

Aprendizaje automático para la detección temprana de la enfermedad de Parkinson 818
Maitane Martinez-Eguiluz, Olatz Arbelaitz, Ibai Gurrutxaga, Javier Muguerza, Iñigo Perona, Ane Murueta-Goyena, Iñigo Gabilondo

Detecting risk of self-harm on Reddit using ensemble traditional and deep learning models 824
Elena Campillo-Ageitos, Juan Martinez-Romo, Lourdes Araujo

The importance of interpretability and visualization in ML for medical applications 830
Alfredo Vellido

Signaleeg:A practical tool for EEG signal data mining 832
Joaquim Massana, Òscar Raya, Jaume Gauchola, Beatriz López

Radiomics textural features extracted from subcortical structures of grey matter probability for Alzheimers disease detection 834
Cesar Ortiz-Toro, Nuria Gutierrez, Consuelo Gonzalo Martin, Roberto Garrido, Alejandro Rodriguez Gonzalez, Ernestina Menasalvas Ruiz

Automatic pneumonia detection in chest X-ray images using textural features 836
César Antonio Ortiz, Angel García-Pedrero, Mario Lillo-Saavedra, Consuelo Gonzalo-Martin

NIO: Neurocognitive Integrated Ontology 838
Alba Gómez-Valadés, Rafael Martínez Tomás, Mariano Rincón

Prediction of COVID-19 severity in Madrid region with Model-leaf trees 840
Miguel A Riaño Sánchez, Concha Bielza, Pedro Larrañaga

Cost-effectiveness analysis with decision analysis networks 846
Francisco Javier Diez, Manuel Luque, Manuel Arias, Jorge Pérez-Martín

I Taller de IA aplicada a los Sistemas Inteligentes de Transporte (IA-SIT)

Cameras vs LiDAR using Deep Learning 851
Fernando Rojas Ramos, Ivo Pineda Torres

Optimización del entorno de simulación para el depurado de arquitectura software en vehículos autónomos . 857
David Yagüe-Cuevas, Alejandro Ruíz-de-la-Cuadra, Pablo Marín-Plaza, José María-Armingol

Deteccion y modelado de la carretera mediante el analisis morfologico de nubes de puntos 863
Alejandro Ruiz-de-la-Cuadra, David Yagüe-Cuevas, Pablo Marín-Plaza, José María-Armingol

Modelado de la carga de vehículos eléctricos basada en agentes 869
Rafael Rus Rus, Agapito Ismael Ledezma Espino

Sistema de detección de fatiga basado en deep learning y lógica difusa 875
Elena Magan Lopez, María Paz Sesmero Lorente, Araceli Sanchis de Miguel

Sistemas empotrados para el reconocimiento cognitivo de conductores 881
Roberto Móstoles Rodríguez, Adrián Losada Casado, Juan Pedro Llerena Caña, Jesús García Herrero, José Manuel Molina López

I Taller de Inteligencia Artificial para el Desarrollo (AI4D)

Teledección híbrida en cultivos leñosos con codificador automático convolucional 889
Iván Rico Prieto, Roberto Hermoso Peralo, Luz Karime Atencia Payares, Jesús Cano Jiménez

Decision Support System for Agricultural Production in Sub-Saharan Africa 895
Bradwell Garanganga, Trymore Nyakutambwa, James Murombedzi, Frank Rutabingwa

Patrones de Circulación Asociados a la Precipitación en la Región de Transición de la Amazonía-Andes 901
Melissa Medina Burga, Eusebio Cisneros, Arturo Sanchez

Machine learning to characterize thefts in Barranquilla City 907
Jhonny Castillo, Paul Sanmartin, Diana Heredia Vizcaino, Daladier Jabba, Elías Said Hung

Fuzzy Logic Based Agent for Flood-Awareness 913
Jose Simmonds, Juan Antonio Gómez, Agapito Ledezma

Fuzzy Logic Approach for Energy Poverty Identification in the Urban Regeneration Areas of Getafe 919
Ascension López Vargas, Raúl Castaño Rosa, Jorge Martínez Crespo, Ulpiano Ruiz-Rivas Hernando, Agapito Ledezma Espino

Clustering energy-poor households using machine learning and socioeconomic indicators 923
Miguel Núñez-Peiró, José Antonio Iglesias Martínez, Ana Sanz-Fernández, Agapito Ismael Ledezma Espino, Carmen Sánchez-Guevara Sánchez, Marta Gayoso Heredia

I Taller de Inteligencia Artificial Cuántica (QUARTI)

Implementable Simple Hybrid Quantum Genetic Algorithm 931
Mikel Garcia-Andoin, Javier Echanobe

Mutual Reinforcement between Neural Networks and Quantum Physics 936
Yue Ban, Javier Echanobe, Erik Torrontegui, Jorge Casanova

A Survey on Quantum Computer Simulators 941
Zakaria Abdelmoiz DAHI, Enrique Alba, Rodrigo Gil-Merino, Francisco Chicano, Gabriel Luque

Quantum Computing: Present and Prospects 947
Rodrigo Gil-Merino, Enrique Alba, Francisco Chicano, Zakaria Abdelmoiz DAHI, Gabriel Luque

I Taller sobre Modelos Generativos Profundos Aplicados (AGM)

GAN-based data augmentation for crack detection 955
Alberto Botana-López, Daniel Gordo-Martín, Adrián Alonso-Rial, Jacobo Otero-Tranchero, Santiago Muiños-Landín

GANs for Urban Design: Learning Subtle Morphological Differences 961
Stanislava Fedorova

I Taller Ciudades inteligentes e Inteligencia Artificial (SCAI)

Citizen Centric Optimal Electric Vehicle Charging Stations Locations in a Full City: Case of Malaga 969
Christian Cintrano, Jamal Toutouh, Enrique Alba

Estudio de la calidad de servicio de autobuses empleando datos abiertos: caso de Melilla 970
Christian Cintrano, José Ángel Morell Martínez, Enrique Alba, Jamal Toutouh

A Collaborative Platform for the Detection of Non-inclusive Situations in Smart Cities 976
Alfonso González-Briones, Iván García-Magariño, Jorge J. Gómez-Sanz, Rubén Fuentes-Fernández, Juan Pavón

Análisis del tráfico de búsqueda de redes WiFi de los Smartphones y sus prestaciones para la monitorización de personas en ciudades inteligentes 977
Antonio Fernández Ares, Antonio Mora, Maribel García Arenas, Pablo García Sánchez, Pedro Castillo

Competición de Apps basadas en Inteligencia Artificial

Recomendación de Restaurantes Basada en Contenido Visual 985
Pablo Pérez-Núñez, Cristina Cuesta Llavona, Jorge Díez

GimmeHop 989
Ignacio Huitzil, Fernando Alegre, Fernando Bobillo

Jizt - AI Summarization 993
Diego Miguel, Mario Juez-Gil, Carlos López-Nozal

KSAS: A Mobile App with Neural Networks to Guide the Learning of Motor Skills 997
Alberto Casas-Ortiz, Olga C. Santos

Sugerencias de Moda 1001
Ismael Pineda-Palencia, Rafael M. Luque-Baena, Miguel A. Molina-Cabello

NBA Analyzer, una app para el apoyo a entrenadores analizando datos de la NBA 1005
Jose Luis Ruiz-Casado, Rafael M. Luque-Baena, Miguel A. Molina-Cabello

Sharpmony: A Computational Intelligence based tool for 4-part harmony 1009
Francisco Fernández de Vega, Jorge Alvarado Díaz, Martín Morita Hernández

BIN-CT 1013
Juan José Guerrero, Javier Ferrer, Rubén Saborido, Enrique Alba

Saru closet 1017
Sara García-Rodríguez, Noelia Rico, Susana Montes, Irene Diaz

Competición de Vídeos de IA Divulgativos

Circuitos que evolucionan 1023
Sara Bueno, Federico Castejón, Enrique Carmona

NAO, el robot cuentacuentos 1026
Félix de La Paz López, Mario Almagro Cádiz, Víctor Fresno Fernández

Doctoral Consortium

Soft Computing and Artificial Intelligence for Interpretable Analysis of Dermoscopic Images 1031
Lidia Talavera-Martínez, Pedro Bibiloni, Manuel González-Hidalgo

Modelado Complejo de Información Lingüística en Problemas de Toma de Decisión en Grupo bajo Incertidumbre 1037
Álvaro Labella Romero, Luis Martínez, Rosa M. Rodríguez

Nuevas aproximaciones de aprendizaje por transferencia en sistemas de recomendación 1041
Eva Blanco-Mallo, Beatriz Remeseiro, Verónica Bolón-Canedo

Algoritmos heurísticos para problemas de embebido de grafos en disposiciones circulares 1047
Sergio Cavero Díaz, Abraham Duarte, Eduardo García Pardo

Modelado y análisis de la marcha humana mediante sensores inerciales y aprendizaje profundo 1053
Diego Teran, Enrique Dominguez

Pseudo-closed elements and closure operators on residuated lattices 1059
Manuel Ojeda-Hernandez

Looking for a Synergy between Boltzmann Machines and Markov Networks 1062
Arkaitz Bidaurrazaga Barrueta, Aritz Perez, Roberto Santana

Banca Conversacional Inteligente: Democratizar los servicios financieros actuales y los nuevos (predictivos, prescriptivos y proactivos) con interacciones más fáciles y naturales que mejoren la Salud Financiera del cliente 1068
Javier Porras Castaño

Taxonomization of combinatorial optimization problems in Fourier space 1074
Anne Elorza

Minería de datos para la sostenibilidad urbana 1079
Francisco Rodríguez Gómez, Llanos Mora López, José del Campo Ávila

PhD Thesis Proposal: Adversarial Machine Learning in the Audio Domain 1085
Jon Vadillo, Roberto Santana, Jose A. Lozano

Artificial intelligence techniques for improving the perception systems of autonomous vehicles 1091
Manuel Carranza-García, Jorge García-Gutiérrez, José C. Riquelme

Design and Implementation of Metaheuristic Algorithms for Social Network Analysis 1097
Sergio Pérez-Peló, Jesús Sánchez-Oro, Abraham Duarte

Optimización heurística de problemas relacionados con redes sociales 1103
Isaac Lozano-Osorio, Abraham Duarte, Jesús Sánchez-Oro

Mathematical Modeling and Analysis of Combinatorial Optimization Problems 1106
Imanol Unanue, María Merino, Jose A. Lozano

Explainable condition monitoring from imprecise information 1110
Nahuel Costa Cortez, Luciano Sánchez Ramos

MORK: Metaheuristic Optimization framewoRK 1116
Raúl Martín Santamaría, José Manuel Colmenar, Abraham Duarte

Índice de autores 1121

XIX Conferencia de la Asociación Española para la Inteligencia Artificial



XIX Conferencia de la Asociación Española para la Inteligencia Artificial

Sesión General



Multi-Objective Bayesian Optimization approach for Environmental Monitoring using an Autonomous Surface Vehicle: Ypacarai Lake Case Study

Federico Peralta
Department of Electronic Engineering
Technical School of Engineering of Seville
Seville, Spain
fperalta@us.es

Samuel Yanes
Department of Electronic Engineering
Technical School of Engineering of Seville
Seville, Spain
syanes@us.es

Daniel Gutiérrez Reina
Department of Electronic Engineering
Technical School of Engineering of Seville
Seville, Spain
dgutierrezreina@us.es

Sergio Toral
Department of Electronic Engineering
Technical School of Engineering of Seville
Seville, Spain
storal@us.es

***Abstract*—The classical Bayesian Optimization approach for optimizing costly and unknown functions is used in this work for monitoring multiple water quality parameters. Since multiple models (one for each water quality parameter) are obtained, we combine BO with a Multi-Objective Optimization method so that optimal measurement locations are determined. These locations are to be visited by an Autonomous Surface Vehicle equipped with water quality sensors. Therefore, we also include adaptations so that the energy constraints of the vehicle are taken into account. The proposed approach delivers multiple surrogate models that prove to be reliable and approximate well to the real models, which in this case, are benchmark functions. For evaluation purposes, we use the R-squared coefficient and compare it with two implementations of water quality monitoring: the Predictive Entropy Search for Multi-Objective Optimization with Constraints algorithm and a Genetic Algorithm path planning for the Ypacarai Lake scenario. The obtained results show that our method is 15.33% better than the former and 22% than the latter, in terms of the R-squared metric with similar average computational times.**

***Index Terms*—Bayes Methods, Multi-objective Optimization, Environmental Monitoring, Path Planning, Autonomous Vehicles, Data acquisition**

I. Introduction

Environmental monitoring using Autonomous Surface Vehicles (ASVs) is an interesting approach nowadays due to their ease of implementation, high scalability, and use of Artificial Intelligence and Machine Learning techniques. Indeed, monitoring a water environment is crucial when the water is polluted or when it is in the process of being treated. Inefficient monitoring will lead to a waste of time and economic resources, and eventually to the loss of such environment. As an example, Ypacarai Lake (Paraguay) is a shallow lake that is currently having periodical blooms of cyanobacteria due to its high polluted state and no large-scale monitoring system [1]. Cyanobacteria is not only toxic for wildlife but also humans. Therefore, the Ypacarai Lake is a great candidate for environmental monitoring, where Water Quality (WQ) parameters, such as pH, total dissolved solids, ion concentration, etc, need to be monitored.

Most ASV-related monitoring systems focus on pure exploration of water bodies [2]. Thus, only a few of them go beyond and manage to intelligently patrol the environment. Furthermore, a small quantity of works takes into account the WQ measures that the system performs to efficiently monitor the water resources. However, the vast majority of these systems only take into account a single parameter, i.e., only one of the many WQ parameters is used for decision-making, despite that the system is usually prepared for performing measures of different parameters at the same time. In this work, we propose a WQ parameter system that uses the data of the multiple available parameters and, through a multi-objective Bayesian Optimization (BO), decides measurement locations for efficient monitoring, hence the name Multi-Objective Bayesian Optimization (MOBO) approach.

BO was chosen due to its model-based design, as it is useful to infer models of the WQ parameters of Ypacarai Lake during the monitoring procedure [3]. Additionally, BO provides an intelligent way of obtaining measurement locations, if the Acquisition Function (AF), or function that

This work has been partially funded by the Spanish "Ministerio de Ciencia, innovación y Universidades, Programa Estatal de I+D+i Orientada a los Retos de la Sociedad" under the Project "Despliegue Adaptativo de Vehículos no Tripulados para Gestión Ambiental en Escenarios Dinámicos RTI2018-098964-B-I00", and by the regional government Junta de Andalucía under the Projects "Despliegue Inteligente de una red de Vehículos Acuáticos no Tripulados para la monitorización de Recursos Hídricos US-1257508", "Despliegue y Control de una Red Inteligente de Vehículos Autónomos Acuáticos para la Monitorización de Recursos Hídricos Andaluces PY18-RE0009", and "Ayuda a Grupo de Investigación PAIDI TIC 201".

evaluates the importance of measuring on a certain location, is appropriately selected. On the other hand, Multi-Objective Optimization (MOO) is used to combine different AFs because it is an optimal decision-making strategy whenever there are numerous objectives, like in this case study. Furthermore, in this work, we make use of different tools and techniques to make monitoring with an ASV possible, such as constraint handling and adaptations of AFs.

The main contributions of this work are the following:

- A MOBO system for WQ monitoring using an ASV.
- A comparison of the MOBO system with other environmental monitoring approaches.

The rest of this work is divided into the remaining sections: Section II introduces the related works, Section III contains the statement of the problem. Next, the proposed approach is found in Section IV, and in Section V a performance evaluation is presented, as well as the discussion of the results. Finally, Section VI presents the conclusions of this work.

II. Related Works

Recent works show that ASVs are used for environmental monitoring [2], [4]–[7]. For example, in [4] a review of autonomous vehicles for monitoring is presented. Several examples show that ASVs are used to increase the efficiency of data collection. In another recent survey [5], the authors remark that ASVs are essential for data acquisition. In [6], the authors proposed to use an ASV for rapid mapping of a WQ parameter using the obtained data for planning. In [7], an ASV obtained a lawnmower-like path planning for optimal coverage of a river, obtaining fast, efficient pure exploration.

The monitoring of the Ypacarai Lake problem was found as a case study as well, where one of the first works using an ASV is found in [2]. They proposed visiting 60 waypoints near the shore of the Lake, and, formulated as a Traveling Salesman Problem, a Genetic Algorithm (GA) was used to find the best visiting order for maximizing exploration. The work showed promising behavior for global path planning. The same authors made variations of the work. For example, in [8], where they compared the work with Eulerian cycles, and in [9] they designed a two-step monitoring process: a general exploration and a focused exploitation step. Deep Reinforcement Learning (DRL) was used in [10] so that an optimal policy for patrolling the Ypacarai Lake was found. A multiple vehicle scenario was also considered in [11], resulting in an efficient decentralized patrolling. A comparison between DRL and GA was also performed in [12]. In [3], the authors proposed the BO approach using a single ASV and a single parameter. The same work laid the foundations for a multi-function estimation using one ASV [13] and for a multi-ASV system [14]. These works [3], [13], [14] have in common that BO was used as the main algorithm, where results show that BO is appropriate for this setting.

For the MOO, in [15] two different kinds of multi-objective methods were used for a path following MPC for vehicle guidance, a weighted sum, and a Pareto optimization. Results show that the latter performed better with fewer evaluations. Exclusively related to MOBO, one approach was proposed in [16]. The authors performed tests using benchmark functions with weighted sums using as arguments Pareto points. The approach performed well and is considered in this work for comparison. In fact, it was used in [13] for a multiple function estimation approach.

Our method differentiates from the related works as it relies on the contribution of several input data, rather than only one objective. Generally, the underlying mission is only exploration [2], [3], [7]. But this work accomplishes both—monitoring and model obtaining—, simultaneously. Our method quickly performs active learning, obtaining approximate models of various WQ parameters, simultaneously. In the next section, we state the problem so that the scope and objective of this work is appropriately defined.

III. Problem Definition

A. Multi-Objective Optimization

Multiple models of WQ parameters of Ypacarai Lake are to be simultaneously obtained using an ASV. This mission will also be accomplished while monitoring the Lake, using BO. Therefore, the proposed approach should take into account the information of multiple functions and generate a solution that maximizes the utility of all the functions, simultaneously. This is also known as the MOO problem, and, for this work, it can be formally expressed as:

$$\begin{aligned} \text{maximize} \quad & \alpha_m(x) && m = 1, 2, \ldots, M \\ \textbf{s.t.} \quad & g_j(x) \leq 0 && j = 1, 2, \ldots, J \\ & x_i^L \leq x_i \leq x_i^U && i = 1, 2, \ldots, I \end{aligned} \tag{1}$$

The previous equation states that there are multiple $\alpha_m(x)$ functions that need to be maximized, but the design space is subject to different constraint functions $g_j(x) \leq 0$ that limit the feasible space. Furthermore, x is an element of the search space, which is bounded by x_i^L lower and x_i^U upper bounds, one for each dimension.

Notice that contrary to common notation found in the literature, we use $\alpha_m(x)$ for the optimization problem, since $f_m(x)$ is defined to denote the real model of a WQ parameter model. Additionally, $\alpha_m(x)$ will correspond to a function that values the importance of performing a WQ measurement in location x. Thus, $\alpha_m(x)$ needs to be function of the real model $f_m(x)$. However, this approach acknowledges that the underlying real models are unknown and need to be approximated with surrogate models. Therefore, it is defined that a surrogate model is approximately equal to the real model plus a noise function $\mathcal{E}^2$, more formally:

$$\hat{f}_m(x) \approx f_m(x) + \mathcal{E}^2 \tag{2}$$

which can be easily obtained. This surrogate model is used to obtain $\alpha_m(x)$, as:

$$\alpha_m(x) = h_m(\hat{f}_m(x)) \tag{3}$$

At this point, it is evident that this MOO problem is not optimizing the set of "main" functions but functions of them.

These functions are to be selected so that WQ monitoring and efficient data acquisition are performed simultaneously.

To further describe the problem and evaluation of the proposed approach, some assumptions and considerations are to be defined.

B. Assumptions and Considerations

The R^2 Score (R2S) is the selected metric that evaluates the goodness or fit of a model. It uses the variance of the real data (defining $\bar{f}_m(x)$) as the average of the real WQ parameter over the search space) and the residual sum of squares using the approximated data:

$$\mathrm{R}^2_m(f_m(x), \hat{f}_m(x)) = 1 - \frac{\sum(f_m(x_i) - \hat{f}_m(x_i))^2}{\sum(f_m(x_i) - \bar{f}_m(x))^2} \quad (4)$$

As multiple functions are approximated, we use the same metric for all the functions and obtain the average R2S of the multiple functions. Next, some additional assumptions and considerations related to WQ environmental systems are described, so that the objective are appropriately defined:

a) Environment: In (1), x was defined as an element of the search space. However, for the vehicle, x corresponds to an (x, y) location in the real world. Using the same notation in (1), the map $\mathcal{M}$ is defined as a rectangular matrix, whose elements correspond to squares where the vehicle can be found. The first element of matrix $\mathcal{M}$ is equal to $(\mathrm{x}^L, \mathrm{y}^L)$ and the last element to $(\mathrm{x}^U, \mathrm{y}^U)$. Therefore, the dimension of the matrix $\mathcal{M}$ is defined by the boundaries and the distance between elements. Furthermore, the elements have an associated value that corresponds to whether location x is a water location, free from any obstacles, reachable by the vehicle. These values are also used to define a function constraint $g_1(x) \leq 0$, in (1). In Fig. 1a, it can be observed that the map has defined values where there exists water, it corresponds to a map used in previous works [3], [13].

b) Navigation, Guidance, and Control (NGC) system: The ASV has an NGC system that provides capabilities for reached the defined destinations. For example, if no direct route between the current location of the vehicle and the goal location is possible, a local path planner, as found in [17], is used so that the vehicle does not collide with obstacles.

c) Water Quality Sensor System: A collection of WQ sensors is used by the vehicle to perform measurements. The number of used sensors determines the m number of models to be approximated with this approach and, consequently, the number of functions that need to be optimized.

d) ASV Constraints: The ASV has limited available energy that allows only for a certain amount of distance traveled. A previous work [3] defined that this maximum distance is $15,000$ (m). Therefore, whenever the vehicle reaches this maximum distance, the vehicle stops and performs one last measurement in that location, then, the mission ends.

IV. Proposed Approach

A. Multi-Objective Bayesian Optimization

MOO was previously defined in Section III as maximizing a set of functions taking into account constraint functions and boundaries. The complete definition can be found in the Subsection IV-D, but first, the BO approach needs to be appropriately described.

BO is a method used for optimizing an unknown function that is costly to evaluate. Therefore, surrogate models are used to alleviate the computational load and to infer the position of optimal sample points. With the use of Acquisition Functions (AFs), locations for measuring the unknown functions are selected. The selection of these locations is balanced to explore and exploit the search space. Then, with the appropriate AFs, monitoring of the search space can occur, while obtaining approximate reliable surrogate models. In this work, BO is based on the next expression:

$$p_{k+1} = \arg\max_x \alpha(x) \quad (5)$$

where p_{k+1} is the next measurement location, which corresponds to the maximum argument of an AF $\alpha(x)$. The AF uses data of the surrogate model and, commonly, that this model is provided by Gaussian Processes (GPs) [18].

B. Gaussian Process

GP is a stochastic process based on multivariate normal distribution [19]. GPs provide regressions using a set of known data and a covariance function. In the case of this work, we define the inputs as the locations in the Ypacarai Lake. Furthermore, a GP is defined as the surrogate model, with regression $\hat{f}(x)$. GPs provide approximate models of real functions, using (2):

$$f_m(x) \approx \mathcal{GP}_m(\mu(x), k(x, x')) \quad (6)$$

This expression defines that every GP consists of a mean value $\mu(x)$ and a covariance function $k(x, x')$.

Covariance function or kernel: The kernel of a function defines the similarities between different inputs and shapes the regression provided by the GP [19]. In this work, we define that the kernel should be smooth and continuous since the expected behavior of WQ parameters is as foretold. Kernel functions take into account not only the unknown inputs but also hyperparameters, that are adjusted during the BO process. In a previous work [3], we performed several tests and found that the Radial Basis Function (RBF) kernel performs the best in these scenarios. RBF has the form of:

$$k(x, x') = \exp(-\frac{||x - x'||^2}{2\ell^2}) \quad (7)$$

where ℓ is the hyperparameter *length-scale*, which defines how far apart can be two different locations in order to influence each other. The value of the ℓ is updated using log marginal likelihood maximization [18], during the fitting process.

GPs can always produce a *prior* model. However, if there exists known data, a *posterior* model can be obtained and a more precise approximate function can be obtained using:

$$\begin{aligned} \mu_{\hat{f}(x_*)|D} &= K_*^T K^{-1} f(x) \\ \sigma_{\hat{f}(x_*)|D} &= K_{**} - K_*^T K^{-1} K_* \end{aligned} \tag{8}$$

The preceding expression [19] takes into account the auto-covariance of the known data input K, the covariance between the known and unknown data input K_* and the auto-covariance of the unknown data input K_{**}. Elements of the different matrices correspond to values of $k(x, x')$, where x and x' can be known or unknown input (locations).

C. Acquisition Functions

With a surrogate model regression for each of the WQ parameters, locations for good measuring must be obtained. AFs are used, as they can conveniently make use of the data provided by the GPs. Additionally, AFs are designed so that they balance exploration/exploitation. For the Ypacarai Lake case study, several AFs were tested in a previous work [3] and the Expected Improvement (EI) performed the best.

EI evaluates how desirable is to perform a new measurement on a location x based on: i) the Cumulative Distribution Function $\Phi(\cdot)$ and ii) the Probability Density Function $\phi(\cdot)$ [18]. The former will help to select supposed optimal locations, while the latter weights the uncertainty of the model. Both take as argument Z, the normalized value of a surrogate model on location x:

$$Z = \frac{f(x^+) - \mu(x) - \xi}{\sigma(x)} \tag{9}$$

The value of EI corresponds the sum of these terms, previously multiplied by exploration exploitation weights:

$$\text{EI}(x) = (f(x^+) - \mu(x) - \xi)\Phi(Z) + \phi(Z)\sigma(x) \tag{10}$$

The weights enhance the decision of performing exploration or exploitation. The parameter ξ is used to balance the exploration/exploitation value. EI provides the desirability of performing measurements anywhere on $\mathcal{M}$.

Furthermore, as the ASV has energy constraints, the vehicle should not travel great distances before performing measurements. To solve this issue, a previous work [3] states that an appropriate manner of performing measurements is to limit the amount of maximum distance allowed from location to location to perform a measurement. The *truncated adaptation* [3] of AF does this, modifying the location of the measurement. The vehicle travels in the direction of the best measurement location according to EI but stops after a defined distance is traveled and measures WQ parameters in that location. The maximum distance is defined by:

$$d = 0.375 \times \ell \tag{11}$$

where ℓ is the characteristic length-scale of the posterior model. This dynamic distance can be defined as the minimum of a set of distances, whenever there are multiple posterior models [13].

We defined the BO procedure as (5) defines: the next measurement location is defined by the maximum argument of an AF. This location is subject to the energy constraint that is defined according to a maximum distance of travel between measurement locations. However, as multiple surrogate models are to be simultaneously approximated, multiple AFs are available, so a MOO is needed.

D. Multi-Objective Optimization

For obtaining an optimal measurement location, a 3–step process, as found in the pymoo API [20], is used:

1) The problem is defined according to (1). The number of available sensors determines the m number of functions to be optimized, the constraint function in this work was already defined by the water locations in the map $\mathcal{M}$. This map also defines the boundaries of search space, i.e., the $g_1(x)$ function.
2) The optimization algorithm is selected. In this work, we use the well-known NSGA-II [21]. It is a multi-objective evolutionary algorithm that sorts non-dominated solutions. Non-dominated solutions correspond to points in the objective space, in which the results are not dominated by each other. GA is used to evolve a set of initial population. Where, according to the procedure shown in [20], the population is evolved so that they become non-dominated solutions via a sorting algorithm.
3) The optimization process is in charge of evolving the non-dominated solutions, according to a defined number of generations. However, since it is necessary to maximize the different AFs, the procedure is done by minimizing the negative AFs, this process is also done in [20].

In this particular MOO case, as the obtained non-dominated solutions are many, we choose the solution that is closer to the position of the ASV, based on the principle that non-dominated solutions do not represent dominance over each other. Therefore, measurements should be performed on any (or all) of these solutions. The selected location is further improved by (11) using the shortest ℓ of the m different fitted GPs.

Finally, the MOBO approach is appropriately defined. In the following section, we evaluate the performance of our proposed method.

V. Performance Evaluation

In this section, the results are presented. The proposed approach was developed in python 3.8.5 and is available online (https://github.com/FedePeralta/BO_drones). For the BO procedure, the Scikit-Optimize (https://scikit-optimize.github.io/) library was used, while for the MOO, the PyMoo [20] library was used. The simulations were run on a Ubuntu Server 18.04, with a 2.24GHz AMD 16-Core Processor and 64GB RAM.

A. Simulation setup

The map corresponds to a 1000 by 1500 black and white image according to the specifications in Section III. One pixel corresponds to a square of side ≈ 10 meters. Therefore, the simulations are run until the simulated ASV travels a total of 1500 pixels.

The Shekel Function (SF) is utilized to generate 2D maps according to the dimensions of the map. The SF is a benchmark function that is multimodal, continuous, and multidimensional, that is, it can have several maxima, there are no discontinuities and it is prepared to work in a 2D environment. SF, in this work, has the form of:

$$f_{\text{Shekel}}(x) = \sum_{i=1}^{M} \frac{1}{c_i + \sum_{j=1}^{2}(x_j - a_{i,j})^2} \tag{12}$$

In this expression, $a_{i,}$ corresponds to a 2D point where a location reaches a maximum value. This value is defined as an inverse importance value c_i, one for each maximum location defined. With the SF, 8 different ground truths maps were generated, with a random number of maxima (between 2 and 6). The different maps have defined values on the pixels that are defined as obstacle-free water squares. An example is shown in Fig. 1a.

Table I shows the definition of the parameters used in the simulations. It is sub-divided into three main categories: i) BO: defines the GPs and AFs, ii) MOO: defines the optimization process using NSGA-II, and iii) MOBO: Uses the GA results to provide a result for (5). The selection of these values was made in previous works and through trial and error.

TABLE I: Simulation parameters.

Parameter		Value
BO	Kernels	RBF ($\ell = 100$)
	AFs	EI ($\xi = 1.0$)
MOO using GA	Optimizer	NSGA-II [21]
	Individual	2 genes: (x, y)
	Population size	50
	Selection	Binary Tournament [20]
	Crossover	Simulated Binary [20]
	Mutation	Polynomial Mutation [20]
	No. Generations.	200
MOBO	Location Selection	Closest Pareto Point
	Adaptation-AF	truncated (using min ℓ_m) [3]

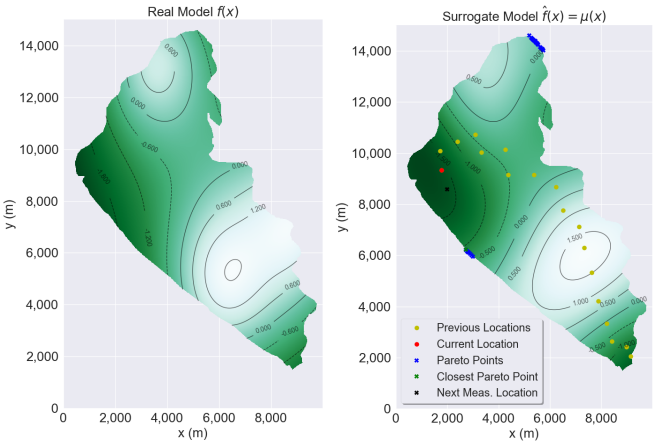
50 different simulations were run for each set shown in Table II. Therefore, the system measured 2, 3, or 4 WQ parameters. Obtaining surrogate models with their respective AFs, which were used to optimize with multiple objectives. Additionally, two different sets of sensors involved were used, so that generalization is also tested. In total, 300 different simulations were run.

B. Simulation results

The proposed MOBO approach accomplishes the mission efficiently. The best simulation result is shown in Fig. 1b, where the Pareto points are shown as blue markers. Notice that, even if the closest Pareto point (green) is selected, the real next measurement location is located at the black marker,

TABLE II: Sets of sensors involved for different simulations.

Sim. ID	Sensors Involved
1	$S1, S2$
2	$S5, S6$
3	$S1, S2, S3$
4	$S5, S6, S7$
5	$S1, S2, S3, S4$
6	$S5, S6, S7, S8$



(a) Example of SF. (b) Example regression of a GP.

Fig. 1: Comparison between (a) the real model and (b) the surrogate model fitted with the known data.

due to the adaptation of AF, i.e., it is truncated relative to the distance to the current location of the ASV (red). The map in Fig. 1b corresponds to the mean of the GP fitted with the known data (yellow and red locations), see (6) and (8). In Fig. 2, an example of the non-dominated solutions in the objective space of two AFs is shown.

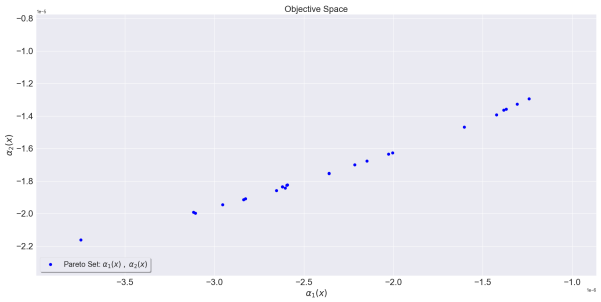


Fig. 2: Example of generated non-dominated solutions.

For the sake of comparison, we implemented the GA [9] and the Predictive Entropy Search MOO with Constraints (PESMOC) Method [16] so that the results of the proposed MOBO approach can be compared. Recall (4) is the expression that shows the R2S, which is the selected metric in this work. In Fig. 3, the results over distance are shown for the three methods, our proposed MOBO approach, the GA and PESMOC, implemented for environmental BO monitoring. It is remarkable how the MOBO approach manages to obtain an

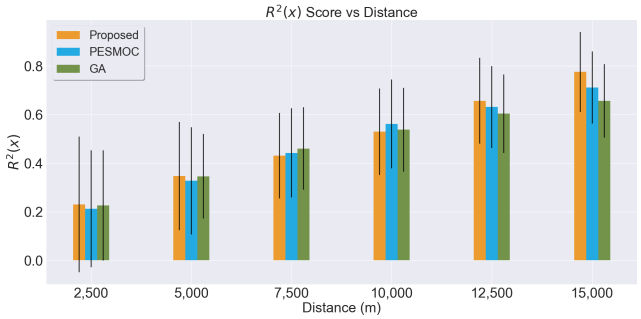


Fig. 3: Average results over distance for the three methods.

average of 0.8193 after 15,000 (m), which is 15.33% better than the PESMOC and 22% than GA. Though the standard deviations are similar, our method presents better results. It is also observable that our method starts outperforming the other methods after 12,500 (m).

On the other hand, the computational time of our method has not outperformed the GA and PESMOC methods (Table III). This is mainly due to the time taken by the NSGA-II optimization process. But for large-scale water environments, like Yparacai Lake, this amount of time is negligible.

TABLE III: Average computational times.

Method	MOBO	GA	PESMOC
Average time (s)	2.9938	1.691	2.8465

VI. Conclusion

In this work, a MOBO approach was proposed for efficiently monitoring an unknown water environment. The system performs measurements of several WQ parameters in locations that will decrease the uncertainty of surrogate models that approximate the real WQ models. These locations are selected using Bayesian Optimization procedures, but instead of selecting the maximum of the AF, the NSGA-II algorithm is used so that multiple AFs are maximized simultaneously, providing non-dominated solutions which are used to find the next measurement location. The locations are adapted so that real-life ASV constraints are taken into account.

The MOBO approach is better than other methods found in the literature. Proving to be an efficient and reliable approach when the mission is to monitor a large-scale water environment, while also measuring its quality parameters. For future work, a multi-ASV multi-objective problem can be addressed, so that reliable surrogate models can be obtained with less time and energy resources.

References

[1] G. A. López Moreira, L. Hinegk, A. Salvadore, G. Zolezzi, F. Hölker, R. A. Monte Domecq S, M. Bocci, S. Carrer, L. De Nat, J. Escribá, et al. Eutrophication, research and management history of the shallow ypacaraí lake (paraguay). *Sustainability*, 10(7):2426, 2018.

[2] M. Arzamendia, D. Gregor, D. Gutierrez Reina, and S. Toral. A path planning approach of an autonomous surface vehicle for water quality monitoring using evolutionary computation. In *Technology for Smart Futures*, pages 55–73. Springer, 2018.

[3] F. P. Samaniego, D. G. Reina, S. Toral, M. Arzamendia, and D. O. Gregor. A bayesian optimization approach for water resources monitoring through an autonomous surface vehicle: The ypacarai lake case study. *IEEE Access*, 9:9163–9179, 2021.

[4] D. O. B. Jones, A. R. Gates, V. A. I. Huvenne, A. B. Phillips, and B. J. Bett. Autonomous marine environmental monitoring: Application in decommissioned oil fields. *Science of the total environment*, 668:835–853, 2019.

[5] B. Bayat, N. Crasta, A. Crespi, A. M. Pascoal, and A. Ijspeert. Environmental monitoring using autonomous vehicles: a survey of recent searching techniques. *Current opinion in biotechnology*, 45:76–84, 2017.

[6] D. P. Nicholson, A. P. M. Michel, S. D. Wankel, K. Manganini, R. A. Sugrue, Z. O. Sandwith, and S. A. Monk. Rapid mapping of dissolved methane and carbon dioxide in coastal ecosystems using the chemyak autonomous surface vehicle. *Environmental science & technology*, 52(22):13314–13324, 2018.

[7] N. Karapetyan, A. Braude, J. Moulton, J. A. Burstein, S. White, J. M. O'Kane, and I. Rekleitis. Riverine coverage with an autonomous surface vehicle over known environments. *arXiv preprint arXiv:1908.02827*, 2019.

[8] M. Arzamendia, I. Espartza, D. Gutierrez Reina, S. Toral, and D. Gregor. Comparison of eulerian and hamiltonian circuits for evolutionary-based path planning of an autonomous surface vehicle for monitoring ypacarai lake. *Journal of Ambient Intelligence and Humanized Computing*, 10(4):1495–1507, 2019.

[9] M. Arzamendia, D. Gregor, D. G. Reina, and S. Toral. An evolutionary approach to constrained path planning of an autonomous surface vehicle for maximizing the covered area of ypacarai lake. *Soft Computing*, 23(5):1723–1734, 2019.

[10] S. Y. Luis, D. G. Reina, and S. L. T. Marín. A deep reinforcement learning approach for the patrolling problem of water resources through autonomous surface vehicles: The ypacarai lake case. *IEEE Access*, 8:204076–204093, 2020.

[11] S. Y. Luis, D. G. Reina, and S. L. T. Marín. A multiagent deep reinforcement learning approach for path planning in autonomous surface vehicles: The ypacaraí lake patrolling case. *IEEE Access*, 9:17084–17099, 2021.

[12] S. Yanes, D. Gutiérrez-Reina, and S. Toral. A dimensional comparison between evolutionary algorithm and deep reinforcement learning methodologies for autonomous surface vehicles with water quality sensors. *Sensors*, 21(8):2862, 2021.

[13] F. Peralta, D. G. Reina, S. Toral, M. Arzamendia, and D. Gregor. A bayesian optimization approach for multi-function estimation for environmental monitoring using an autonomous surface vehicle: Ypacarai lake case study. *Electronics*, 10(8), 2021.

[14] F. Peralta, S. Yanes, D. G. Reina, and S. Toral. Monitoring water resources through a bayesian optimization-based approach using multiple surface vehicles: The ypacarai lake case study. In *Accepted for IEEE Congress on Evolutionary Computation, CEC 2021, Kraków, Poland, June 28-July 01, 2021*. IEEE, 2021.

[15] A. Gharib, D. Stenger, R. Ritschel, and R. Voßwinkel. Multi-objective optimization of a path-following mpc for vehicle guidance: A bayesian optimization approach. *arXiv preprint arXiv:2104.03773*, 2021.

[16] E. C. Garrido-Merchán and D. Hernández-Lobato. Predictive entropy search for multi-objective bayesian optimization with constraints. *Neurocomputing*, 361:50–68, 2019.

[17] F. Peralta, M. Arzamendia, D. Gregor, D. G. Reina, and S. Toral. A comparison of local path planning techniques of autonomous surface vehicles for monitoring applications: The ypacarai lake case-study. *Sensors*, 20(5):1488, 2020.

[18] F. Archetti and A. Candelieri. *Bayesian optimization and data science*. Springer, Berlin, Germany, 2019.

[19] C. E. Rasmussen. Gaussian processes in machine learning. In *Summer school on machine learning*, pages 63–71, Berlin, Germany, 2003. Springer.

[20] J. Blank and K. Deb. Pymoo: Multi-objective optimization in python. *IEEE Access*, 8:89497–89509, 2020.

[21] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE transactions on evolutionary computation*, 6(2):182–197, 2002.

Genetic Enhanced Model-based Deep Reinforcement Learning for Informative Path Planning in the Lake Ypacaraí Scenario

Samuel Yanes Luis
dept. of Electronic Engineering
Sevilla Higher School of Engineering
Sevilla, España
syanes@us.es

Federico Peralta Samaniego
dept. of Electronic Engineering
Sevilla Higher School of Engineering
Sevilla, España
fperalta@us.es

Daniel Gutiérrez-Reina
dept. of Electronic Engineering
Sevilla Higher School of Engineering
Sevilla, España
dgutierrezreina@us.es

Sergio Toral Marín
dept. of Electronic Engineering
Sevilla Higher School of Engineering
Sevilla, España
storal@us.es

***Abstract*—The patrolling of large water resources is a strenuous task when the traditional manual sampling methods are applied. The use of Autonomous Surface Vehicles equipped with water quality sensors serves the purpose of monitoring those zones of high interest for the authorities and biologists. In a particular application of Artificial Intelligence, the Non-Homogeneous Informative Path Planning is solved in the Ypacaraí Lake using a Model-Based Deep Reinforcement Learning algorithm to synthesize, in a sample-efficient way, a behavioral policy for the vehicle to follow. This approach is based on a genetic-synthesized model of the inner dynamics which, assuming some hypothesis of the lake, ensures diversity of experiences to enhance the exploration. The model-based approach achieves 57% better solutions in the online simulations with an order of magnitude fewer training episodes. Furthermore, the proposed algorithm is able to perform the training 50% faster, which encourages its use in critical-time applications like the Ypacaraí Lake patrolling case.**

***Index Terms*—Informative Path Planning, Autonomous Surface Vehicle, Deep Reinforcement Learning, Environmental Patrolling**

I. Introduction

The Ypacaraí Lake is the bigger water surface of Paraguay faces a major contamination problem because of the industrial and touristic continuous development of near cities in recent years. The lack of waste disposal infrastructure causes the waters to raise their levels of phosphates, nitrates and suspended solids, which has a dramatic consequence in the form of blue-green algae blooms all over the water surface. These colonies of cyanobacteria cause a serious impact on the biological equilibrium of the environment because the water becomes toxic for the local fauna and also for humans. As a result of the algae blooms, a fetid smell comes off the waters affecting the attraction of what was once a great source of tourism.

To solve this problem, the very first task is to obtain a model of the environmental variables of the lake. This contamination map will serve the purpose of tracing the most interesting zones to watch over and to close to the public bath. This task can be arduous if not impossible when we consider the manual sampling of the Ypacaraí waters, which has more than $60km^2$. Therefore, in this work as in previous ones [1]–[5], it is considered the use of Autonomous Surface Vehicles (ASVs) to automate the acquisition and monitoring missions of water-quality variables. The application of ASVs to such tasks allows a more energy-efficient monitoring and considerable time saving because of the autonomy of such vehicles. Nonetheless, these vehicles must face complex tasks and require planning of the movements in two different scales: i) the local path planning and ii) the global path planning. In the former, obstacle avoidance and point-to-point planning are attended. In the latter, a much more complex task is pursued, that is, what physical points are worth traveling in order to optimize certain criteria.

In this particular environmental mission, Informative Path Planning (IPP) is addressed. This means that, given a certain map of interest, which can be a map of the contamination peaks of the lake, the global path planning must focus on guiding the ASV to visit those zones which maximize the information gathered. The information takes the form of water-quality data taken in certain areas. The water-quality map of information has been synthesized in previous works using a

This work has been partially funded by the Spanish "Ministerio de Ciencia, innovación y Universidades, Programa Estatal de I+D+i Orientada a los Retos de la Sociedad" under the Project "Despliegue Adaptativo de Vehículos no Tripulados para Gestión Ambiental en Escenarios Dinámicos RTI2018-098964-B-I00", and by the regional government Junta de Andalucía under the Projects "Despliegue Inteligente de una red de Vehículos Acuáticos no Tripulados para la monitorización de Recursos Hídricos US-1257508" and "Despliegue y Control de una Red Inteligente de Vehículos Autónomos Acuáticos para la Monitorización de Recursos Hídricos Andaluces PY18-RE0009".

Bayesian Optimization approach for the very same ASVs [4], [5]. Once it is obtained, the path planner must solve the optimal trajectory to collect the maximum available information without any prior knowledge of the map and avoiding any collisions. In order to conform to the dynamic behavior of the algae colonies, it is significant to consider also a redundancy criterion, where high-information zones must be revisited after a while, in periodic missions. Hence, an Informative Non-Homogeneous Patrolling (INHP) is pursued in the Ypacaraí monitoring.

It is remarkable how the task of patrolling with an arbitrary information map is a very complex problem, which falls within the domain of NP-Hard problems. These kinds of problems must be addressed using meta-heuristics methodologies or, in this case, an Artificial Intelligence approach such as Deep Reinforcement Learning (DRL). DRL algorithms are useful when the environment dynamics are hard to model and a decision problem is addressed. Convolutional Neural Networks (CNN), permit such algorithms to infer optimal movements by a reactive policy in global path planning from a graphical representation of the scenario. Nevertheless, it has an important drawback: the lack of sample efficiency. As the agent in this particular DRL algorithm Double Deep Q-Learning (DDQL) optimizes its policy using trial and error by interacting several times with the environment, it makes DDQL not applicable directly for online purposes because of the high number of interactions needed. In order to overcome this issue, a model-based approach is implemented. If the environment is known to a certain point, a train-by-proxy optimization could speed up the learning and obtain good policies when deployed online.

In summary, this work presents the well-known DRL technique called Double Deep Q-Learning using a novel model data-based enhancement of the agent knowledge in order to pursue a sample-efficient online implementation. In this sense, the diversity of experiences must be ensured and be efficient to guarantee to the DRL algorithm a sufficient diversity of interactions to learn from. For this purpose, a Genetic Algorithm with Niching (GA-N) [6] mechanism provides enough diversity in those experiences to optimize the Neural Q-Network of the aforementioned algorithm.

Hence, the contributions of this work are:

- A diversity-enhanced genetic generation of model experiences to train the deep agent.
- A model-based Double Deep Q-Learning approach for online INHP in the Ypacaraí scenario.

The rest of the paper is organized as follows. In Section 2, a brief of the state of the art of DRL and its application in ASVs are presented. Secondly, Section 3 defines the scenario and the different assumptions taken into account for the deployment. In section 4, the algorithm is presented. In section 5, the results of simulations and hyper-parameters configurations are discussed. Finally, the conclusions of this work are explained in Section 6.

II. State of the art

The use of ASVs for environmental missions has attracted the attention of researchers because of the recent developments in autonomy and energy-efficiency [3], [7], [8]. In the particular case of monitoring natural environments, these vehicles have resulted extremely beneficial due to their small size and the ability to take samples of the environmental variables, such as pH, turbidity, and temperature, among others [9]. In order to obtain a contamination map, remarkable advances can be found in [5] and [4], where the Bayesian Optimization is applied for single and multiple ASVs in the Ypacaraí scenario. Other works focus on the path planning problem itself, addressing different goals, for instance, the energy-consumption constraints [10] or the complete coverage using a lawn-mower approach [11].

In the particular case of Informative Path Planning (IPP), recent researches like [12], [13] have presented important results for unmanned vehicles. In [12], an online Dynamic Programming (DP) method for an ASV is proposed to collect samples and survey the environmental information for modeling the temperature of a navigable area. In [13], a similar approach is used but with a DRL approach. The information entropy is minimized in a Double Deep Q-Learning framework (like in our work) using a dynamic directed graph.

It is very frequent in the literature to find DRL applied to such Path Planning tasks similar to the one here addressed. In [14], for example, a Deep Q-Learning application to unmanned vehicles for the total coverage path planning can be found. In [8], an Actor-Critic architecture is used for training a multi-shape autonomous robot with several degrees of freedom to fulfill an NP-Hard coverage task. Those works have in common that the problem has been solved off-line with numerous interactions with the environment because the scenario is known a priori. This is not possible in the target problem of the present work; and therefore, an online application must be pursued.

On the Genetic Algorithms side, there are many examples in the literature of the application of such methods for path planning. In [15], for example, the full-coverage task of the Ypacaraí Lake is addressed using an online approach to intensively cover new algae blooms. Other works, like the ones described in the survey [16], combine the Reinforcement Learning approaches with Evolutionary Algorithms to improve the performance in both directions: RL used to improve GAs and vice versa. An example of the former can be found in [17], where RL helps with parameter control for continuous evolutionary optimization. The latter case can be found in [18], where GA helps Q-Learning to perform an online multi-objective optimization. The main difference between our work and the aforementioned ones is that, in our case, the genetic algorithm helps to generate experiences to train with the DDQL assuring diversity in an online deployment. Those other works, are usually limited to an off-line optimization or do not consider the sample efficiency in their results, only the convergence.

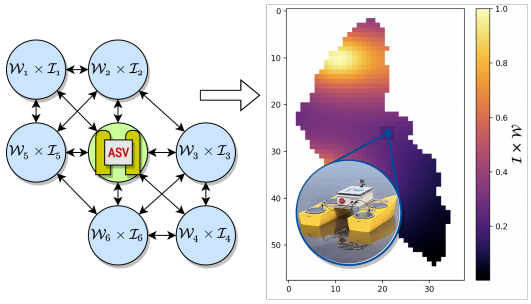


Fig. 1. Modeling of the Non-Homogeneous Informative Patrolling Problem into a discrete interest map of the Ypacaraí Lake. In the left, an image of the ASV prototype used in the real-case implementation.

III. Methodology

This section is organized as follows: first, the NHIPP is explained within the assumptions of the developed simulator. Next, the GA-N is explained. Finally, the Enhanced DDQL algorithm is presented.

A. Statement of the Problem

For the NHIPP in the Ypacaraí Lake, a metric and undirected graph $G(E, V, \mathcal{I}, \mathcal{W})$ can be defined in order to divide the physical map into discrete nodes V. Every node/zone is connected with its surroundings which can be visited by performing an action $a \in [N, S, E, W, NE, NW, SE, SE]$ (see 1 for a graphical conceptualization of the modeling). In this graph it is defined also an information value $I \in [0, 1]$ in every node, and a idleness level $W \in [0, 1]$ also. The information value refers to the amount of information in this place, which ponders how important is it to visit this node. The idleness level refers to the time the node has remained unsampled. Every $\mathcal{I}$ is weighted with its idleness value $\mathcal{W}$ to obtain the Relative Information Value (RIV) $\tilde{\mathcal{I}} = \mathcal{I} \times \mathcal{W}$. This will balance the informative value of every node and the temporal redundancy of visitations. For example, a high informative zone could be not so worth visiting if it has been recently covered than a zone of lower importance but unvisited yet. It is also specified that a zone is considered visited when it enters the circular detection area $\mathcal{M}$ of the ASV.

This way, it is defined a reward function $r(s_t, a_t)$ that will measure the adequacy of every action a_t, given the state of the graph s_t in a particular timestep t. The

$$r(s_t, a_t) = \begin{cases} -10, & \text{if } a_t \text{ is illegal.} \\ \sum_{(x,y)\in\mathcal{M}} \left[\frac{\mathcal{I}(x,y)\times\mathcal{W}(x,y)}{|\mathcal{M}|}\right], & \text{otherwise.} \end{cases} \tag{1}$$

The reward function (1) benefits those movements that cover zones within the detection area with higher information interest and higher waiting times. In order to normalize the reward, $\tilde{\mathcal{I}}$ is divided by the detection area (a constant value). When an action is illegal because intends to move towards the land, a penalization is applied. In the end, the optimization problem is to maximize the obtained information along a path of a particular length, which is, to maximize the Accumulated Episodic Reward (AER):

$$\begin{gathered} \text{maximize } AER = \textstyle\sum_{t=0}^{t=T} r(s = s_t, a = a_t) \\ \text{subject to } G(V, E, \mathcal{W}, \mathcal{I}, \mathcal{M}) \end{gathered} \tag{2}$$

A simulator has been designed to test the proposed algorithm. Some assumptions have been also made:

- Every episode begins in the same geographical place (see Figure 1).
- The length of the path will be 30 km, according to the battery duration of the ASV, at full speed (2 m/s)
- The possible movements are $a \in A = \{N, S, E, W, NE, NW, SE, SO\}$.
- When an illegal action is intended, no movement is performed, but when training, the corresponding penalization is applied.
- In order to simulate possible uncertainties and fails of the inner control loop of the ASV autopilot, there is always a 5% probability to perform a random action instead of the chosen one.

B. Double Deep Q-Learning

The DDQL algorithm is a Reinforcement Learning technique that tries to solve a Markov Decision Problem (MDP) which is a mathematical framework for sequential decision dilemmas. It has been defined previously the possible actions $a \in A = \{N, S, E, W, NE, NW, SE, SO\}$ that the agent (ASV) can perform given a state $s \in \mathcal{S}$ of the problem. As the state-action domain $\mathcal{S} \times \mathcal{A}$ is extremely large, the Double Deep Q-Learning approach tries to estimate an action-sate value function $Q(s, a)$ which translates a state into the expected value of the accumulated and discounted reward in the future for every possible action. Being γ a discount factor to ponder the importance of the long-term rewards:

$$Q(s, a) = \mathbb{E}\left(\sum_{k=0}^{N} \gamma^k r_{t+k}(s_k, a_k)\right) \tag{3}$$

This function is clearly impossible to estimate using brute force, and DDQL uses a learn-by-trial approach to learn the relation between s and r through a. In this case, the Q-function will be a Convolutional Neural Network (CNN) that will try to fit its values with the, also estimated, target temporal difference. The target temporal difference is defined as the maximum expected discounted reward for the next state. In the DDQL, the target Q-function $Q^*(s, a; \theta')$ is a clone of the behavioral policy Q-function $Q(s, a; \theta)$, which updates its parameters θ' with the weights of the behavioral Q-function θ gradually. Hence, the target temporal difference Y_t is:

$$Y_t = r_t + \gamma \max_{a' \in A} Q^*(s_{t+1}, a'; \theta'_t) \tag{4}$$

This leads to a better convergence because it reduces the overestimation in the greedy operator in (4). By collecting experiences (s_t, a_t, r_t, s_{t+1}) into a Memory Replay Buffer

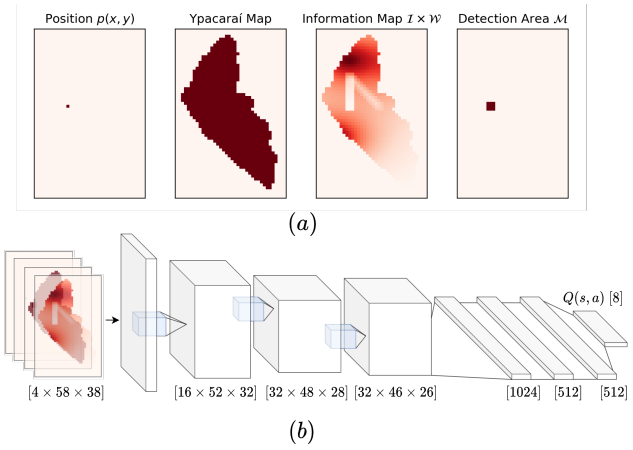


Fig. 2. In (a), the four channels of the state: the position, the boundaries map, the information map and the detection area. In (b), the Convolutional Neural Network architecture. All the layers have a ReLU activation function except for the last one, which has none.

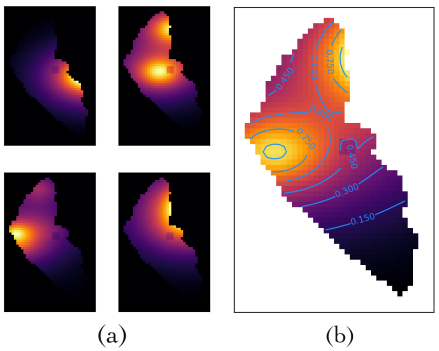


Fig. 3. In (a), four random information maps generated in the model. In (b), the information map in the real scenario, not known a priori and only used for evaluation purposes. The model uses a family of Shekel functions with random peaks like in [5], as a coherent contamination level distribution.

(MRB), $Q(s,a;\theta)$ is optimized using a gradient-descent update of a loss function $\mathcal{L}$ given a mini-batch $\mathcal{B}$ of random past experiences

$$\theta_{t+1} = \theta_t + \alpha \frac{\partial}{\partial \theta_t} \mathcal{L}\left(Y_{\mathcal{B}}, Q(s \in \mathcal{B}, a \in \mathcal{B})\right) \quad (5)$$

with α as the learning rate. This optimization process is, in fact, very sample-inefficient because thousands of experiences must be collected in order to get the function to converge. This limits the online application to deterministic environments where the scenario is known after the training, like in [3].

In this work, the state will be represented as an image of four channels with all the necessary information of the environment (see Figure 3a). The state is processed by the CNN following the architecture in Figure 3b. The convolutional part extracts important features by applying the convolution operation and, the last fully-connected part synthesizes the eight possible values of Q. This particular architecture has been tested in previous works with successful results.

C. Niching Genetic Algorithm

DDQL takes benefits from the diversity of interactions between the agent and the environment. Classically, an explorative-greedy policy called ϵ-greedy is used to choose random actions with an ϵ probability and exploit greedily the knowledge, using $a = \max_{a'} Q(s,a')$ with $(1-\epsilon)$ probability. The MRB must be enriched with sufficiently diverse experiences (s_t, a_t, r_t, s_{t+1}) to ensure the Q-function is learned unbiased. Nonetheless, a random exploration usually does not ensure the best paths to be part of the experiences. In this sense, Genetic Algorithms (GAs) could serve the purpose of optimizing a model of the environment to generate experiences similar enough to a real environment, since they are very explorative algorithms. For this application to succeed, it is important that the GA is able to maintain diversity over the individuals, since the MRB must contain bad decisions and good ones. In this sense, it is proposed a Genetic Algorithm with Niching mechanism [6] to ensure diversity of solutions in the paths of a model of the real case. This model tries to mimic the dynamics of the a priori unknown information distribution of the Lake.

The GA-N algorithm consists of a generic GA that selects in an elitist way their individuals by tournament from a generation to another. The individuals are defined as a list of possible actions of a length equal to the maximum movement size. Those individuals are selected to mutate with a $P(Mutate)$ probability and to crossbreed with a $P(Cross)$ probability. In this particular Niching modification, the fitness of every individual is weighted depending on how similar is to the others. By penalizing in the elitist selection process those solutions that are very similar to another. An Average Hamming Distance Operator [1] is used to measure the distance between solutions. As a result, the algorithm can obtain good paths and also maintain diversity in the population.

Once the GA-N is executed for the model, we can observe in Figure 4 how the average AER grows in both cases, but in the Niching case, the average distance between individuals is always bigger. It is also remarkable how the bad paths are still present which is convenient too for the DRL. This indicates this algorithm is suitable to provide diverse solutions for pre-training and online optimization.

D. GA-Enhanced DDQL

In this work, a DDQL algorithm is applied to solve the NHPP focusing on an online application of the algorithm. For that purpose, the architecture depicted in Figure 5 is implemented in simulation. First, the GA-Niching optimizes the model in a pre-training offline process in two steps: i) after N generations, the individuals (paths) in the population will fill a Genetic Experience Memory. ii) from the Genetic

[1]The Avg. Hamming Distance is, in this case, the XOR operation between two individuals, gene by gene. Therefore, two identical individuals have 0 distance and vice versa.

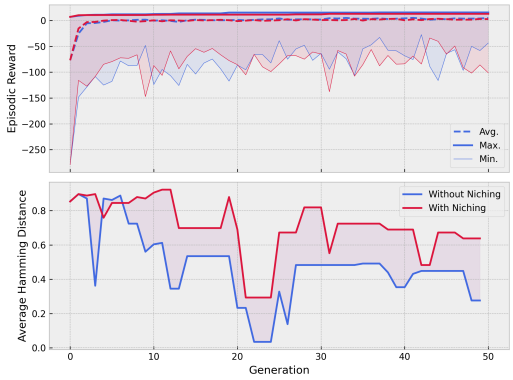


Fig. 4. Results of applying the Niching to the GA.

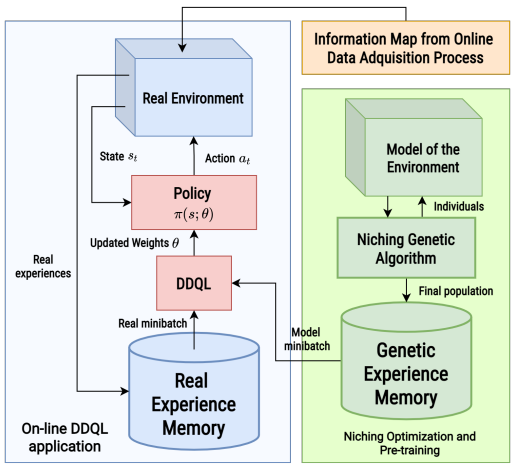


Fig. 5. GA-Enhanced DDQL for the on-line deployment of the ASV.

TABLE I
SIMULATION HYPERPARAMETERS.

Hyperparameter	Value
Number of generations N_{Gen}	50
Population size $\|Pop\|$	300
Genetic Buffer size	5000
Real Buffer size	2000
Genetic Batch Size $\|P_m\|$	32
Real Batch Size $\|P_r\|$	32
Genetic Batch Learning Rate α_g	1×10^{-4}
Real Batch Learning Rate α_r	1×10^{-3}
Pretraining Genetic Iterations K_{Pre}	1000
Training Online Genetic Iterations K_{On}	1

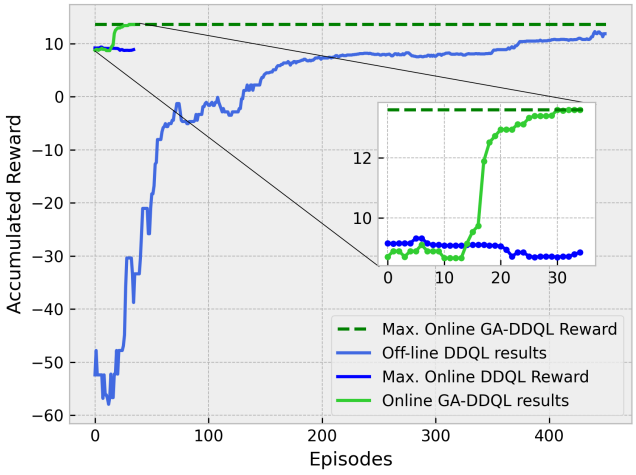


Fig. 6. Training results for the propposed approach. In green, the average accumulated reward for the online deployment of the GA-DDQL. In blue, the results of using an offline trained DDQL.

Experience Memory, P_m gradient-descent steps are taken for $Q(s, a; \theta)$ by uniformly sampling $|P_m|$ experiences. Once the $Q(s, a; \theta)$ function is pre-trained using only the genetic offline experiences, the ASV can be deployed.

It is assumed that previously to the beginning of the first patrolling mission, the ASV acquires a $\mathcal{I}$ based on expert knowledge or using a Bayesian Optimization approach as it was mentioned in Section I [4], [5]. When the modeling mission ends, the patrolling begins, and the GA-DDQL is activated. For every movement, a mini-batch P_r of real experiences is sampled and a descent-gradient step is performed as in a DDQL. Furthermore, to enhance the training based on the previous model experiences, a mini-batch P_m is also sampled and a gradient descent-step is taken. This way, the DDQL can learn from both memories. The GA experiences are more accessible than the real interactions and can be generated in parallel, in spite of not being a perfect representation of reality.

IV. RESULTS

In order to test the proposed methodology, it is simulated the real interaction of the ASV with the environment using only one ground truth. Every mission has a duration of 4 hours at full speed (2 m/s). For the simulation, 50 missions are simulated, which limits the full patrolling campaign to 25 days (two missions per day). It is assumed also, in order to validate this novel approach, that there are no significant changes in the real environment from a mission to another. Other parameters of the simulation are summarized in Table I.

The results of the simulation can be seen in Figure 6. This shows a comparison between a classic off-line DDQL approach (light-blue line) with the online algorithm with GA pre-training. It can be seen how the number of episodes required to obtain similar levels of the average reward is greater by an order of magnitude without the GA-pretraining.

It is also noticeable, apart from the sample-efficiency metric, how the maximum achieved reward is superior when compared to the application of the classic DDQL transferred to the online deployment. As the algorithm is continuously trained with the genetic-model experiences and the real experiences as well, the performance of the patrolling after 20 episodes is a 42.4 % better than the classic DDQL. At the end of the patrolling campaign, the GA-DDQL obtains a 53% improvement over its classic counterpart. These results also prove the proposed

TABLE II
COMPUTATIONAL TIMES FOR GA-DDQL AND DDQL

	GA-DDQL	**Basic DDQL**
Gradient-Descent Steps	6.000	40.000
Pre-training Time	9.23 mins	14.5 mins
Computation time per step	0.25 s	0.15 s

solution responds well to the cold-start problem every DDQL algorithm has: as the training depends on the generated experiences when no previous experiences are available, the learning starts irremediably slow and inapplicable to an online scenario.

Regarding the computational efficiency, in Table II is the most important computational times and metrics for this application. The number of gradient-descent steps in the model pre-training is 6.67 times less which proves the Genetic Enhanced method is more sample-efficient in the selection of the useful experiences. Regardless of the sample efficiency of the proposed methodology, it is important to notice that, as the GA-DDQL optimizes online also using the genetic memory, the computation time of the next action is higher.

V. CONCLUSIONS

In this work, the Non-Homogeneous Informative Path Planning is pursued from an online perspective in the Lake Ypacaraí using Autonomous Surface Vehicles. In order to solve this problem, a Double Deep Q-Learning algorithm has been proposed with a novel modification using the Genetic Algorithm with Niching. This approach puts the focus on obtaining diverse experiences to enrich a genetic buffer from a model of the environment to train the Convolutional Neural Network when the access to real experiences has a high cost. The results of the simulation have validated the efficiency of this algorithm when compared to its more classical DDQL variant. The proposed algorithm performs better in a fewer number of training episodes. Regarding the computational times, as the GA can be executed in parallel for the pretraining, it obtains better wall times. I

In further steps, this algorithm should consider the mutation of the model according to the online samples. The use of a model of the lake has resulted vital in order to boost up the learning efficiency. In this sense, an adaptive scheme should be tested where the model updates itself from the real data in order to have a complete online adaptive learning process. It is also interesting to perform an ablation study to determine how the diversity of the genetic buffer intervenes in the learning process. As there are many variations in Evolutionary Algorithms, it is interesting to validate the contribution to the learning of the Niching process and how different operators affect the final performance.

REFERENCES

[1] S. Yanes Luis, D. Gutiérrez-Reina, and S. Toral Marín, "A dimensional comparison between evolutionary algorithm and deep reinforcement learning methodologies for autonomous surface vehicles with water quality sensors," *Sensors (Switzerland)*, vol. 21, no. 8, p. 2862, 2021.

[2] S. Yanes Luis, D. G. Reina, and S. L. T. Marin, "A Deep Reinforcement Learning Approach for the Patrolling Problem of Water Resources Through Autonomous Surface Vehicles: The Ypacarai Lake Case," *IEEE Access*, vol. 8, pp. 204 076–204 093, 2020.

[3] ——, "A Multiagent Deep Reinforcement Learning Approach for Path Planning in Autonomous Surface Vehicles: The Ypacaraí Lake Patrolling Case," *IEEE Access*, vol. 9, pp. 17 084–17 099, 2021.

[4] F. P. Samaniego, D. G. Reina, S. L. T. Marín, M. Arzamendia, and D. O. Gregor, "A bayesian optimization approach for water resources monitoring through an autonomous surface vehicle: The ypacarai lake case study," *IEEE Access*, vol. 9, pp. 9163–9179, 2021.

[5] F. Peralta, D. G. Reina, S. Toral, M. Arzamendia, and D. Gregor, "A bayesian optimization approach for multi-function estimation for environmental monitoring using an autonomous surface vehicle: Ypacarai lake case study," *Electronics*, vol. 10, no. 8, 2021. [Online]. Available: https://www.mdpi.com/2079-9292/10/8/963

[6] C.-G. Lee, D.-H. Cho, and H.-K. Jung, "Niching genetic algorithm with restricted competition selection for multimodal function optimization," *IEEE transactions on magnetics*, vol. 35, no. 3, pp. 1722–1725, 1999.

[7] J. Sánchez-García, J. M. García-Campos, M. Arzamendia, D. G. Reina, S. L. Toral, and D. Gregor, "A survey on unmanned aerial and aquatic vehicle multi-hop networks: Wireless communications, evaluation tools and applications," *Computer Communications*, vol. 119, pp. 43–65, apr 2018.

[8] A. Krishna Lakshmanan, R. Elara Mohan, B. Ramalingam, A. Vu Le, P. Veerajagadeshwar, K. Tiwari, and M. Ilyas, "Complete coverage path planning using reinforcement learning for tetromino based cleaning and maintenance robot," *Automation in Construction*, vol. 112, no. May 2019, p. 103078, 2020. [Online]. Available: https://doi.org/10.1016/j.autcon.2020.103078

[9] H. Ferreira, C. Almeida, A. Martins, J. Almeida, N. Dias, A. Dias, and E. Silva, "Autonomous bathymetry for risk assessment with roaz robotic surface vehicle," in *Oceans 2009-Europe*, 2009, pp. 1–6.

[10] B. Liu, Y. Zhang, S. Fu, and X. Liu, "Reduce UAV coverage energy consumption through actor-critic algorithm," *Proceedings - 2019 15th International Conference on Mobile Ad-Hoc and Sensor Networks, MSN 2019*, pp. 332–337, 2019.

[11] I. A. Hameed, "Coverage Path Planning Software for Autonomous Robotic Lawn Mower using Dubins' Curve," in *IEEE International Conference on Real-time Computing and Robotics*, IEEE, Ed. Okinawa, Japan: 517-522, 2017.

[12] R. Mishra, M. Chitre, and S. Swarup, "Online informative path planning using sparse gaussian processes," *2018 OCEANS - MTS/IEEE Kobe Techno-Oceans (OTO)*, pp. 1–5, 2018.

[13] Y. Wei and R. Zheng, "Informative Path Planning for Mobile Sensing with Reinforcement Learning," *Proceedings - IEEE INFOCOM*, vol. 2020-July, pp. 864–873, 2020.

[14] M. Theile, H. Bayerlein, R. Nai, D. Gesbert, and M. Caccamo, "UAV Coverage Path Planning under Varying Power Constraints using Deep Reinforcement Learning," 2020. [Online]. Available: http://arxiv.org/abs/2003.02609

[15] M. Arzamendia, D. Gutierrez, S. Toral, D. Gregor, E. Asimakopoulou, and N. Bessis, "Intelligent online learning strategy for an autonomous surface vehicle in lake environments using evolutionary computation," *IEEE Intelligent Transportation Systems Magazine*, vol. 11, no. 4, pp. 110–125, 2019.

[16] M. M. Drugan, "Reinforcement learning versus evolutionary computation: A survey on hybrid algorithms," *Swarm and Evolutionary Computation*, vol. 44, no. April 2017, pp. 228–246, 2019. [Online]. Available: https://doi.org/10.1016/j.swevo.2018.03.011

[17] G. Karafotias, M. Hoogendoorn, and A. E. Eiben, "Evaluating reward definitions for parameter control," in *Applications of Evolutionary Computation*, A. M. Mora and G. Squillero, Eds. Cham: Springer International Publishing, 2015, pp. 667–680.

[18] K. Van Moffaert, M. M. Drugan, and A. Nowé, "Hypervolume-based multi-objective reinforcement learning," in *Evolutionary Multi-Criterion Optimization*, R. C. Purshouse, P. J. Fleming, C. M. Fonseca, S. Greco, and J. Shaw, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 352–366.

Bi-objective Evolutionary Approach for Water Resource Patrolling Problem with Autonomous Surface Vehicles

Samuel Yanes
Electronic Engineering Department
Universidad de Sevilla
Seville, Spain
syanes@us.es

Federico Peralta
Electronic Engineering Department
Universidad de Sevilla
Seville, Spain
fperalta@us.es

Daniel Gutiérrez-Reina
Electronic Engineering Department
Universidad de Sevilla
Seville, Spain
dgutierrezreina@us.es

Alejandro Tapia
Engineering Department
Universidad Loyola Andalucía
Seville, Spain
atapia@uloyola.es

Álvaro Rodriguez
Electrical Engineering Department
Universidad de Sevilla
Seville, Spain
arnozal@us.es

Sergio Toral
Electronic Engineering Department
Universidad de Sevilla
Seville, Spain
storal@us.es

***Abstract*—Monitoring water resources is of paramount importance for the maintenance of natural ecosystems, including fauna and human being. Climate change, which is mainly caused by human bad practices in daylife and industrial activities, is putting in danger the quality of water bodies such as rivers, lakes, and reservoirs, among others. Therefore, new monitoring systems should be proposed for efficient and effective management of these key infrastructures. Taking into consideration the large extension and the dynamic of the water resources, autonomous vehicles could play an important role in this monitoring task. In this paper, a bi-objective global path planner based on NSGA-II with variable chromosome representation for autonomous surface vehicles is proposed. The proposed global path planner maximizes the covering of zones of high interest according to a contamination model, while minimizing the distance travelled by the vehicles. The proposed approach has been validated in a simulated scenario of the Ypacarai Lake in San Bernardino, Paraguay.**

***Index Terms*—Genetic Algorithms, Multi-objective optimization, Water Monitoring, ASV, Patrolling Problem**

I. Introduction

Water monitoring is a crucial task for the maintenance of nature and the well-being of humans. In addition, water resources such as lakes, rivers, lagoons and reservoirs, among others, are key infrastructures for water supply in agriculture, cattle raising, and human and animal consumption. However, climate change, mainly due to the human intervention, is putting at risk the quality of water masses. The uncontrolled disposal of contamination agents and the greenhouse effect, which is causing an increase in the temperature and level of the water, are clear examples of these threats. Therefore, it is necessary to propose monitoring systems for managing and protecting the water resources effectively. Traditionally, the monitoring has been carried out by a manual sampling of water and a posterior analysis in a laboratory. Furthermore, fixed stations are also placed in some locations of water resources to sample water quality parameters. This allows to monitor water masses in real-time and remotely by connecting these systems with the Internet and the cloud. Nevertheless, manual and fixed stations present important limitations, such as the cost of manual sampling and the large extension of water resources which make inefficient the utilization of fixed emplacements. As a consequence of the mentioned drawbacks, Autonomous Surface Vehicles (ASVs) equipped with sensors have been recently proposed for monitoring tasks [1], [2].

ASVs are small and low cost ships, when compared with the normal ones, that can navigate autonomously without human intervention. The idea of water monitoring through a fleet of ASVs consists of equipping the vehicles with sensors that can measure parameters related to the water quality, such as pH, conductivity, dissolved oxygen, and temperature, among others. The mobility of ASVs (2-3 m/s) enables to extend the monitoring mission throughout the whole water resource efficiently, and develop water quality models with the data samples [3], [4] and using machine learning approaches. However, the main constrain of ASVs is that they are supplied by batteries; therefore, they can only function during several hours. Thus, the mobility planning of the ASVs should be made taking into consideration such mobility restrictions.

This work has been partially funded by the Spanish "Ministerio de Ciencia, innovación y Universidades, Programa Estatal de I+D+i Orientada a los Retos de la Sociedad" under the Project "Despliegue Adaptativo de Vehículos no Tripulados para Gestión Ambiental en Escenarios Dinámicos RTI2018-098964-B-I00", and by the regional government Junta de Andalucía under the Projects "Despliegue Inteligente de una red de Vehículos Acuáticos no Tripulados para la monitorización de Recursos Hídricos US-1257508" and "Despliegue y Control de una Red Inteligente de Vehículos Autónomos Acuáticos para la Monitorización de Recursos Hídricos Andaluces PY18-RE0009".

Global path planners are the algorithms that determine the set of waypoints or interest points that the ASV should visit [5]. Normally, after reaching a target waypoint the ASV takes a water sample and continues with the following one in the set.

Path planning is a complex task (NP-hard) that depends on the intended monitoring mission, such as covering or exploring water resources [2], tracking contamination spots, and/or patrolling whenever a contamination model is known [6]. Due to the complexity of the path planning problem, heuristic approaches are normally used like Genetic Algorithms (GAs) [7] and Deep Reinforcement Learning (DRL) approaches [6], [8]. In this paper, a bi-objective path planning algorithm is proposed based on NSGA-II algorithm [9]. The target monitoring task is the patrolling problem of the Ypacarai Lake in San Bernardino in Paraguay, which is suffering from eutrophication of the water due to the uncontrolled disposal of phosphorus and nitrate. The target objectives are i) the coverage maximization of the interesting zones and ii) the minimization of the distance travelled by the ASVs since it is related to the power consumption. In order to generate variable routes with different length and messy GA is employed [10]. Furthermore, the target monitoring problems is solved in two modes: i) a single ASV is used and ii) a fleet of ASVs is employed. The main contributions of this work can be summarized as:

- A Bi-objective path planning algorithm for patrolling problem of water resources based on NSGA-II algorithm with variable chromosome length.
- The validation of the proposed approach in the Ypacarai lake for two modes of operation: single-ASV and multi-ASV cases.

This paper continues as follows: Section II includes related works of monitoring approaches based on ASVs. The statement of the problem is described in Section III. Section IV details the proposed approach based on NSGA-II algorithm and Section V contains the simulation results obtained in a simulated scenario of the Ypacarai lake. Finally, the main conclusion of the paper can be found in Section VI.

II. Related work

Several works have addressed the utilization of ASVs for water resource monitoring in the last few years [2], [3], [6], [7], [11], [12]. The main artificial intelligence techniques used to determine the global path planning of the vehicles are: Genetic Algorithms (GA) [2], [12], Deep Reinforcement Learning (DRL) [6], Swarm Intelligence (SI), and Bayesian Optimization [3]. Regarding the use of GAs, in [2], the authors solve the global path planning problem by modelling it as the well-known Travelling Salesman Problem (TSP). In contrast to the classical problem, in [2], the authors maximize the distance since it is related to the covered area. This work is extended in [12], where the authors model the problem as Chinese Postman Problem (CPP). Thus, the covering problem is transformed into finding the optimum Eulerian cycle to maximize the distance travelled instead of the optimum Hamiltonian cycle searched by the TSP modelling. Although the complexity of the problem is increased, the CPP modelling outperforms the TSP counterpart. With regard to DRL methods, in [6] a Double Deep Q-learnig (DDQ) approach is proposed to solve the Non-homogeneous patrolling problem in the Ypacarai Lake. The authors test several neural networks architectures for the problem and carry out the hyper-parametrization of the networks. The obtained results are promising in terms of convergence and achieved rewards. In [13], the previous work is extended by using a multi-agent approach with centralized learning, that is, a neural network to control the movements of the whole fleet of vehicles. A novel neural network architecture is proposed for a DDQ approach that is able to scale well for moderate number of vehicles (3-5 vehicles). The authors compare their centralized approach with respect to the independent Q-learning, where each vehicle is controlled by a different network. The proposed centralized learning shows significant superiority in terms obtained training time consumption and results (achieved reward). A comparison between evolutionary and deep reinforcement approaches can be found in [8]. The authors carry out a dimensional evaluation and comparison between genetic algorithm and DRL techniques for the patrolling problem of Ypacarai lake. The DRL approach outperforms the evolutionary technique as the complexity of the problem increases. However, the genetic algorithm exhibits good performance in terms of dispersion, showing very similar results for each iteration and a lower sensibility to the hyper-parameters of the algorithm.

Bayesian optimization has been recently proposed for Informative Path Planning (IPP) [3] [4]. It is important to highlight that IPP problem is a different monitoring task with respect to the previous covering and patrolling problems addressed by GAs and DRL since the main objective is to efficiently sample a water resource to obtain a model of the parameters related to quality of the water. In [3], the authors adapt classical Bayesian optimization for the mobility and sampling restrictions of an ASV in a monitoring mission. The obtained results demonstrate the Radial Basic Function (RBF) kernel and Expected Improvement (EI) acquisition function are suitable for the smooth scenarios expected in water monitoring applications. In [4], the work is extended by considering several quality parameters simultaneously.

Some works that pursue the same general objective as this work can be found also in the literature [14]–[16]. In [14], Ant Colony Optimization is used to define the locations of multiple agents in a multi-objective scenario, where worst idleness, energy consumption and communication idleness are to be optimized. The different agents obtain individual strategies for minimizing their costs. Other works, such as [15], [16], define different approaches of Evolutionary Algorithms for patrolling objectives. The former manages to design patrolling strategies based on vulnerability strategies for border security, while the latter designs an algorithm for maritime patrolling using short feedback systems.

This works presents a step forward with respect previous work regarding some aspects. First, previous approaches such

as [2] [7] do not applied a multi-objective methodology, since they are only focused on one objective. Similarly, the comparison presented in [8] do not consider a multi-objective approach. Second, this work presents a variable length representation or chromosome of individuals. It allows to have a high flexibility about the movements of vehicles.

III. Statement of the Problem

This section describes the target monitoring problem. For this purpose, we consider the Ypacarai Lake as the water resource to be monitored by one or several ASVs equipped with water-quality sensors. A contamination map of one water-quality parameter is considered to have been obtained previously. For this task, the readers are referred to the work in [3]. The Lake is considered to be discretized in equal size cells. Every cell is reachable by a vehicle from the adjacent ones if it is navigable; therefore, the cell is a water zone. The non-visitable zones correspond to land areas and they are considered as obstacles. Using the mentioned discretized version of the lake, it is possible to define a graph $G(E, V, W)$ where every reachable area or cell corresponds to a node V. The edges E represent the cost of navigating from one cell to another, resulting in a metric graph. A W is a metric related to the visitation frequency of a node, which is called idleness in this work.

A. The Non-Homogeneous Patrolling Problem

In this work, the Non-Homogeneous Patrolling Problem (NPP) is considered. In this type of PP, each cell of the lake has a different importance or interest according to the contamination level. As a rule, the higher the contamination, the higher the importance of the cell. Therefore, an importance term $\mathcal{I} \in [0, 1]$ for every cell is assigned. Furthermore, since such important zones should be revisited in order to track the contamination levels, an idle value W for every cell is also assigned. This metric describes the timesteps since the vehicle's last visit. The main idea behind W is that the vehicles revisit important zone periodically. Considering $\mathcal{I}$ and W, the NPP can be mathematically defined as:

$$\mathcal{P}^* := \{V_1, .., V_N\} \rightarrow \max_P \sum_{i=1}^{N} \left(\frac{\mathcal{W}_i}{\mathcal{W}_{max}} \times \mathcal{I}_i \right) \qquad (1)$$

Where N is the total number of movements of the vehicle and $\mathcal{W}_{max}$ is the maximum idle value for a cell. Notice that in the case of a fleet with multiple vehicles N takes into account the movements of all vehicles.

B. Bi-objective Non-Homogeneous Patrolling Problem

The optimization problem of the NPP is extended by considering also the distance travelled by the ASVs. The main idea is to minimize the distance travelled by the ASVs $d_{total} = \sum d_i$, since it is related to required power consumption. Therefore, the resulting bi-objective optimization problem can be defined as:

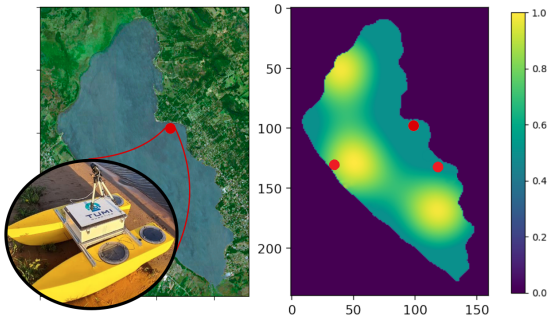


Fig. 1. At the left, the Ypacaraí map with three deployment zones (red dots). At the right, the continuous importance function of the scenario. Higher values denotes more importance for the zone, like higher contamination levels, more probability of an algae bloom, etc.

$$\begin{aligned} &\max P, \min d_{total} \\ &s.t. \quad d_{total} <= d_{max} \end{aligned} \qquad (2)$$

Where d_{max} is the maximum distance that the vehicle can travel, which is dependent on the battery autonomy of the vehicles. Additionally, it has been specified, in order to balance the level of coverage of the lake between visited and unvisited zones, that every visit will decrease the importance $\mathcal{I}_\rangle$ of this particular zones in a 20%. This means that a very important zone (x, y) with $\mathcal{I}_{\S,\dagger} = 1$, will only contribute to the fitness in the five first visits until its value becomes zero.

C. Assumptions on vehicles movements

For the evaluation of the proposed approach the following assumptions for the vehicle movements and the monitoring mission are made:

- The vehicles can perform 8 different actions a. Every action, if possible, is a movement that commands the agent to move to one of the adjacent cells, which are defined as $a \in A := \{N, S, E, W, NE, NW, SE, SW\}$. In addition, diagonal movements have an equivalent cost of the square root of two of a perpendicular movement.
- It is not possible to travel to an adjacent land cell. This action is penalized following the fitness function described in the next section.
- There are three initial areas for the vehicles to deploy that corresponds to the accessible permitted zone of the lake (red dots in Fig. 1). These zones correspond (east to west) to Playa Chini (Areguá), Costa Serena and Playa de San Bernardino (City of San Bernardino). These zones are of high touristic interest and optimal places for deployment.
- An ASV can travel along 39.5 km at full speed (2 m/s) and with full charge (plus a safety minimum for the return to land). This distance will be translated to cells depending on the map resolution.
- To simulate possible fails in the control loop that governs the actuators of the ASV, it is imposed a 5% probability of randomness in every action taken.

IV. Proposed Bi-objective evolutionary approach

The proposed bi-objective path planner is based on the well-known NSGA-II algorithm [9] and messy GA [10]. GA are evolutionary algorithms based on Darwinian theory that allow to evolve a population of potential solutions, which are encoded in a chromosome-like structure, which are called as individuals. The evolution is conducted through genetic operations, such as selection, crossover and mutation.

In multi-objective problems, where more than one objective should be optimized simultaneously, Pareto dominance is normally employed to determine the Pareto Front and the Pareto set, that is, the solutions that are not dominated. Furthermore, messy GA [10] are evolutionary approaches where the length of the chromosomes is not fixed, this allows more flexibility in the generated solutions. In the following subsections, the different elements of the bi-objective GA used are defined.

A. Individual representation

The chromosomes are formed by the movements conducted by the ASVs. In the multi-ASV case, the chromosomes contain the movements of all vehicles. Fig. 2 contains the individual representations for both cases under study. Notice that M_i stands for the movements i of an ASV and N is the total number of movements. The value of N is limited according to the battery lifetime of the ASV. A movement M_i corresponds to one of 8 defined actions $a \in A := \{N, S, E, W, NE, NW, SE, SW\}$. It is important to highlight that in both cases the lengths of the chromosomes are variable. In addition, in the case of the multi-ASV system, the length of the chromosome is divided by the number of vehicles so that each vehicle performs the same number of movements N.

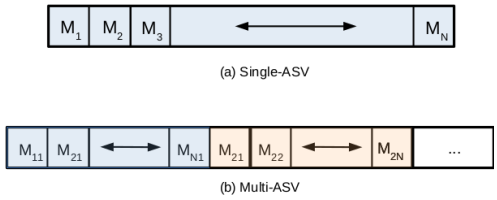


Fig. 2. (a) Individual representation for (a) the single-ASV case and (b) the multi-ASV case.

B. Fitness function

The proposed bi-objective optimization problem tries to optimize two objectives simultaneously: i) the Non-homogeneous patrolling problem given by (1) and ii) the total distance travelled by the ASVs. It is important to highlight that in the multi-agent case there can be collisions among vehicles. To avoid this undesirable solution, a penalization factor is added to (1). Therefore, the bi-objective functions employs is:

$$\max \sum_{i=1}^{N} \left(\frac{\mathcal{W}_i}{\mathcal{W}_{max}} \times \mathcal{I}_i \right) - (\mathcal{C} \times \mathcal{P}_{col}) - (\mathcal{K} \times \mathcal{P}_{mov}), \min d_{total} \quad (3)$$

It is important to recall that N is the total number of movements of all vehicles, I_i and W_i are the interest and idleness values of each cell respectively and W_{max} is the maximum idleness value for a cell. C is defined as the number of collisions among vehicles if multiple vehicles are used and P_{col} as the penalization factor (see more details Section V). Notice that if a vehicle tries to move to a land cell it is also considered as a collision. In addition, the length of the chromosomes is controlled by penalizing patrolling fitness as long as it gets lager than the maximum number of movements allowed. For this purpose, the parameter K accounts for the excess of movements and P_{mov} is the penalization factor (see more details in Section V). Finally, d_{total} is the total distance travelled by all ASVs.

C. Genetic operators

The selection mechanism used is the tournament with a size of 3. It has been demonstrated that this size is suitable for many problems [17]. The crossover technique used is the *messy-one-point* mechanism, since it allows to work with chromosomes with variable length. Regarding the mutation scheme, the shuffle indices technique is used. Finally, the stopping criterion of the GA is the maximum number of generations.

V. Performance evaluation

This section contains the simulation-based validation of the proposed approach. The simulation settings are first described, including the simulated scenario of the Ypacarai Lake with a contamination model of a single water quality parameter and the hyper-parameters used in the GA. Secondly, the simulation results obtained are presented. A simulator has been developed in Python using DEAP library [18], the code is available in a Github repository[1].

A. Simulation settings

Table I contains the simulations parameters used in the simulations. For the single-ASV case only one vehicle is used, while in the multi-ASV case, up to three ASVs are used. It is relevant to highlight that for each case the Pareto front obtained is the result of running the NSGA-II with up to four different combinations of crossover and mutation probabilities. Then, the resulting Pareto front is obtained by merging the results of each run in a final Pareto front. Thus, the proposed approach is able to achieve a higher diversity of individuals in the Pareto front. The values of I, P_{col} and W_{max} has been selected according to [6]. The value of P_{mov} has been demonstrated to be sufficient to avoid invalid solutions.

In Fig. 3 it is represented the importance map $\mathcal{I}$ for this particular work. The importance map is, as it was mentioned before, a discretization of a continuous information scalar field. This map is modelled attending to different criteria, including biological interest and expertise knowledge, among others. For this work, the interest map has been selected using the very same ground truth as in [4]. Therefore, a Shekel

[1] https://github.com/Dany503/Bi-objective-Patrolling-Problem

TABLE I
SIMULATION PARAMETERS.

Parameter	Value
Max. chromosome length	50 (single-ASV), 150 (multi-ASV)
Min. chromosome length	2 (single-ASV), 6 (multi-ASV)
Nº generations	400
Nº individuals	2000
Crossover	Messy one point
Mutation	Shuffle indices
Crossover probability P_c	$[0.8, 0.7, 0.6, 0.5]$
Mutation probability P_m	$[0.2, 0.3, 0.4, 0.5]$
Selection	Tournament with $size = 3$
Collision penalization P_{col}	5
Excess of movements penalization P_{mov}	3
Interest values I	$[0, 1]$
Maximum idleness W_{max}	50

function with three peaks will model the interest of the zones with a minimal interest of 0.5 to encourage the exploration of not visited zones.

The resolution of the discretization of the Shekel function will affect on the size of the individuals, the convergence and complexity of the problem, following the conclusions of [8]. A resolution of 0.5 km^2 for every cell is selected. This resolution has been proven sufficient in order optimize the problem in an affordable amount of time and to obtain an adequate convergence [13].

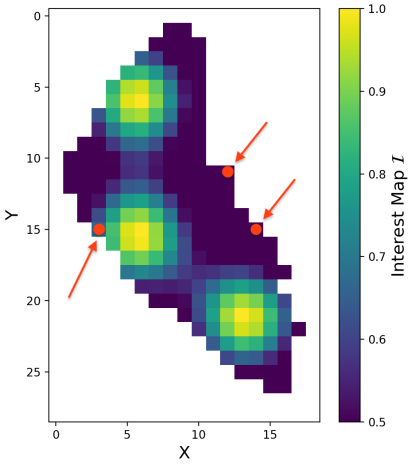


Fig. 3. Discrete Ypacaraí importance map. This map represents graphically $\mathcal{I} \in [0, 1]$. The brighter the zones, the higher the biological or touristic importance for the patrolling task.

B. Simulation results

Fig. 4 and Fig. 5 show, respectively, the Pareto fronts obtained for the single-ASV and multi-ASV cases. In the single-ASV case, it can be observed that the ASV requires the maximum allowed distance in order to achieve suitable results. For short distances, the ASV focused on a single peak of the objective function. Therefore, for the utilization of a single ASV the battery autonomy should be enlarged.

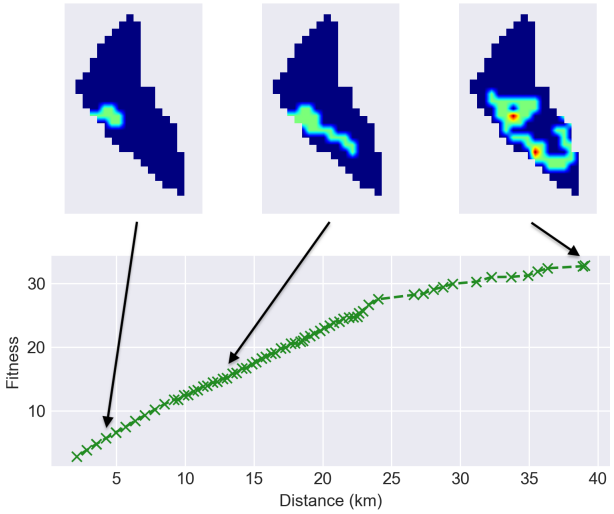


Fig. 4. Pareto front resulting of the single-agent bi-objective optimization (down). In the top, three heat-maps of the resulting movements of particular solutions.

In the multi-ASV case, the ASVs can share the patrolling task, and consequently, the results are better. It can be observed that the ASVs visit the three peaks of the target function. In this case, it is not required to reach the maximum allowed distance to patrol the Ypacarai Lake.

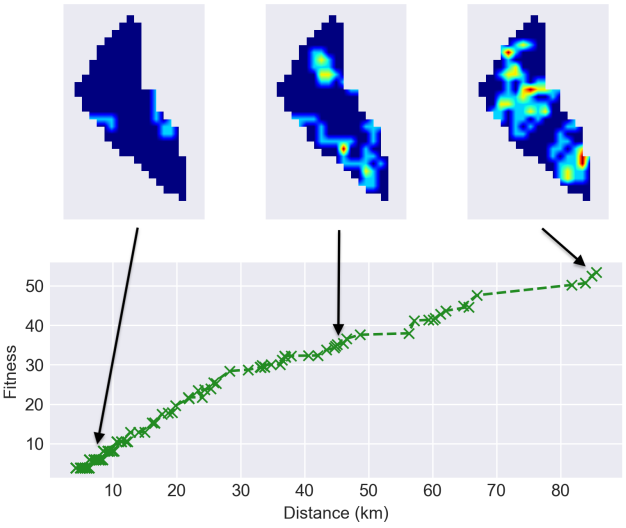


Fig. 5. Pareto front resulting of the multi-agent bi-objective optimization (down). In the top, three heat-maps of the resulting movements of particular solutions.

In both cases, a lineal relationship between the patrolled area and the distance travelled by the ASVs can be seen.

TABLE II
INDIVIDUAL FITNESS OBTAINED FOR EVERY AGENT FOR THREE PARETO SOLUTIONS.

Distance	Agents	Fitness
7.9 km	ASV_1	19.41
	ASV_2	19.13
	ASV_3	14.89
45 km	ASV_1	10.16
	ASV_2	11.57
	ASV_3	12.02
85.6 km	ASV_1	2.0
	ASV_2	3.81
	ASV_3	2.0

After a given point, a saturation effect can be observed. This happens because of the specifications of the fitness function: as the areas are visited numerous times, the total possible interest decreases. When the paths become very large, the maximum achievable fitness becomes tighter. This effect is specially important in the multi-agent case, where the number of visits of every zone is higher as there are multiple vehicles.

As a result of the optimization, given the initial points in Fig. 3, it is possible to obtain the best place to deploy an ASV. This place, in $(x, y) = (3, 15)$, which corresponds to the Playa Chini deployment point, has been observed to be the initial point for the single-agent Pareto dominant solutions. This approach then, can serve the purpose of providing the best initial point of deployment among a set of possible candidates.

In regards of the cooperative behavior in the multi-ASV case, in Table II it can be seen the individual fitness obtained by every ASV for the three Pareto solutions of Fig. 5. It is remarkable how the fitness is quite balanced between the three individuals. This situation is desirable as the patrolling task does not falls on a particular ASV, but it is shared by the fleet. The Pareto solution for the biggest path, indicates that the ASVs that are closest to the top two peaks of interest, will cooperate to cover this zone, and the third ASV (the lower in the right according to Fig. 1), is in charge of the lowest interest peak.

VI. CONCLUSIONS

This paper proposes a bi-objective global path planner for ASVs aimed at monitoring missions. The approach is based on the well-known NSGA-II technique with messy representation of individuals. The monitoring task consists of the NPP of a water quality parameter. The system has been validated in a simulated scenario based on Ypacarai Lake in Paraguay. The simulation obtained results demonstrate the goodness of the proposed approach, including the best initial points for the ASVs and a balance in the workload in the multi-ASV case. As future work, the proposed bi-objective approach will be extended by considering several water quality models simultaneously. Thus, the ASVs should move to maximize the NPP for each water quality parameter. In addition, further analyses about the penalty factors for both the collisions and the maximum distance will be conducted.

REFERENCES

[1] J. Sánchez-García, JM. García-Campos, M. Arzamendia, DG. Reina, SL. Toral, and D. Gregor. A survey on unmanned aerial and aquatic vehicle multi-hop networks: Wireless communications, evaluation tools and applications. *Computer Communications*, 119:43–65, apr 2018.

[2] M. Arzamendia, D. Gregor, DG. Reina, and SL. Toral. Evolutionary path planning of an autonomous surface vehicle for water quality monitoring. In *2016 9th International Conference on Developments in eSystems Engineering (DeSE)*, pages 245–250. IEEE, 2016.

[3] F. Peralta Samaniego, D. G. Reina, S. L. Toral Marín, Derlis O. Gregor, and M. Arzamendia. A bayesian optimization approach for water resources monitoring through an autonomous surface vehicle: The ypacarai lake case study. *IEEE Access*, 9:9163–9179, 2021.

[4] F. Peralta, D. G. Reina, S. Toral, M. Arzamendia, and D. Gregor. A bayesian optimization approach for multi-function estimation for environmental monitoring using an autonomous surface vehicle: Ypacarai lake case study. *Electronics*, 10(8):963, 2021.

[5] N. Wang and H. Xu. Dynamics-constrained global-local hybrid path planning of an autonomous surface vehicle. *IEEE Transactions on Vehicular Technology*, 69(7):6928–6942, 2020.

[6] S. Yanes, D. G. Reina, and S. L. Toral Marín. A deep reinforcement learning approach for the patrolling problem of water resources through autonomous surface vehicles: The ypacarai lake case. *IEEE Access*, 6:1–1, 2020.

[7] Mario Arzamendia, Derlis Gregor, Daniel Gutierrez Reina, and Sergio Luis Toral. An evolutionary approach to constrained path planning of an autonomous surface vehicle for maximizing the covered area of ypacarai lake. *Soft Computing*, 23(5):1723–1734, 2019.

[8] S. Yanes Luis, D. G. Reina, and S. Toral. A dimensional comparison between evolutionary algorithm and deep reinforcement learning methodologies for autonomous surface vehicles with water quality sensors. *Sensors*, 21(8), 2021.

[9] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE transactions on evolutionary computation*, 6(2):182–197, 2002.

[10] D. E. Goldberg, B. Korb, and K. Deb. Messy genetic algorithms: Motivation, analysis, and first results. *Complex systems*, 3(5):493–530, 1989.

[11] L. Bottarelli, M. Bicego, J. Blum, and A. Farinelli. Orienteering-based informative path planning for environmental monitoring. *Engineering Applications of Artificial Intelligence*, 77:46–58, 2019.

[12] M. E. López Arzamendia, I. Espartza, D. G. Reina, S. L. Toral, and D. Gregor. Comparison of eulerian and hamiltonian circuits for evolutionary-based path planning of an autonomous surface vehicle for monitoring ypacarai lake. *Journal of Ambient Intelligence and Humanized Computing*, pages 1495–1507.

[13] S. Yanes, D. G. Reina, and S. L. Toral Marín. A multiagent deep reinforcement learning approach for path planning in autonomous surface vehicles: The ypacaraí lake patrolling case. *IEEE Access*, 9:17084–17099, 2021.

[14] E. T. Fute, E. Tonye, F. Lauri, and A. Koukam. Multi-agent patrolling: Multi-objective approach of the event detection by a mobile wireless sensors network. *International Journal of Computer Applications*, 88(12), 2014.

[15] M. Muaafa and J. E. Ramirez-Marquez. Bi-objective evolutionary approach to the design of patrolling schemes for improved border security. *Computers & Industrial Engineering*, 107:74–84, 2017.

[16] L. Huang, M. Zhou, and K. Hao. Non-dominated immune-endocrine short feedback algorithm for multi-robot maritime patrolling. *IEEE Transactions on Intelligent Transportation Systems*, 21(1):362–373, 2019.

[17] S. Luke. *Essentials of Metaheuristics*. Lulu, second edition, 2013. Available for free at http://cs.gmu.edu/∼sean/book/metaheuristics/.

[18] F. Fortin, F. De Rainville, M. Gardner, M. Parizeau, and C. Gagné. DEAP: Evolutionary algorithms made easy. *Journal of Machine Learning Research*, 13:2171–2175, jul 2012.

Sistema Inteligente para la Gestión Eficiente de Recogida de Residuos Plásticos

Juan José Guerrero*, Javier Ferrer†, Rubén Saborido†, Enrique Alba†
*Lenguajes y Ciencias de la Computación, Universidad de Málaga, España
†ITIS Software, Universidad de Málaga, España
Email: jjguerrero@lcc.uma.es, ferrer@lcc.uma.es, rsain@uma.es, eat@lcc.uma.es

***Resumen*—La falta de optimización en la gestión de la recogida de residuos sólidos en nuestras ciudades es un problema creciente para nuestra sociedad. No obstante, para paliar esta situación ya existe tecnología que puede ser aplicada. En este artículo proponemos BIN-CT, un sistema de gestión de recogida de residuos basado en inteligencia artificial que se sustenta en tres pilares. 1) Novedosos contenedores equipados con sensores y un sistema de compactación para extender el tiempo útil entre vaciados, 2) software que utiliza los datos emitidos por los sensores para realizar predicciones del nivel de llenado y 3) algoritmo evolutivo que genera rutas eficientes diarias para el vaciado de los contenedores. Con el fin de validar los beneficios del sistema hemos realizado tres experimentos: (i) comparamos el sistema tradicional de recogida con nuestra propuesta, (ii) medimos la tolerancia de nuestro sistema a la pérdida de datos de los sensores y (iii) comparamos la planificación de rutas diseñada por expertos y las generadas por BIN-CT. Los resultados muestran que nuestra propuesta reduce tanto emisiones contaminantes como costes en comparación a los medios tradicionales usados actualmente en la mayoría de las ciudades.**

***Index Terms*—*Deep Learning*, Predicción, *Vehicle Routing Problem*, Reciclaje, Sostenibilidad**

I. Introducción

En los últimos tiempos, nuestro planeta ha experimentado tanto un crecimiento demográfico significativo, así como una revolución tecnológica sin precedentes que se traducen en una demanda de producción cada vez mayor. Esta creciente demanda ha causado que la generación de residuos y el control de la polución se conviertan en grandes problemas que, actualmente, no son atendidos en la medida que se necesita [1], [2]. Estos problemas tienen sus raíces principalmente en el modelo lineal de consumo en el que se basa nuestro modelo económico. Los bienes son producidos, consumidos y desechados, generando grandes cantidades de residuos durante el proceso [3].

Al abordar tareas tan complejas como la gestión de los residuos, la eficiencia es uno de los factores clave. No obstante, ni autoridades locales ni entidades privadas suelen invertir suficientes recursos para mejorar la recogida de residuos, donde se suele aplicar siempre el mismo plan estático en la mayoría de los casos [4]. Este enfoque es muy ineficiente, dado que los contenedores dentro de una misma ruta pueden sufrir cambios drásticos en su ritmo de llenado a través del tiempo. Una posible mejora al plan estático es registrar manualmente los niveles de llenado de los contenedores gracias a los conductores [5], aunque esta opción es muy propensa a errores. Por tanto, el uso de sensores para monitorizar periódicamente el nivel de llenado de cada contenedor nos va a permitir descubrir patrones en el llenado de forma fiable. Aun así, la decisión sobre qué contenedores deben ser recogidos cada día sigue siendo enormemente complicada.

Aunque trabajos previos ya utilizan diferentes técnicas para calcular rutas eficientes minimizando tiempo, costes y emisiones contaminantes, no se suele tener en cuenta el nivel de generación de residuos en cada contenedor para esta tarea [6], [7]. Esto puede deberse a lo poco que se ha experimentado con contenedores que integren sensores de llenado [8], [9].

En este artículo proponemos BIN-CT, un sistema de predicción basado en Inteligencia Artificial (IA) y tecnología IoT (Internet of Things) para la gestión eficiente de recogida de residuos plásticos. Nuestro sistema se apoya en una red de contenedores sensorizados para conocer su llenado y equipados con un novedoso sistema de compactación para aumentar su capacidad efectiva, es decir, para incrementar el tiempo entre recogidas. Esto implica que las compactaciones realizadas desde la última recogida son un factor a tener en cuenta cuando calculamos la capacidad restante de un contenedor y tratemos de predecir su evolución. Estas predicciones son fundamentales para decidir qué contenedores deben ser recogidos y que serán la entrada del algoritmo evolutivo que genera rutas de recogida eficientes.

Como ayuda a la validación de nuestra propuesta, planteamos las siguientes preguntas de investigación (*Research Questions* - RQs) que serán respondidas a lo largo del artículo:

RQ1. *¿Cuales son los beneficios reales de utilizar contenedores compactadores sensorizados?*
Esta RQ nos permite cuantificar los beneficios de utilizar contenedores compactadores sensorizados respecto al uso de contenedores tradicionales.

RQ2. *¿Como afecta el número de medidas del llenado de los contenedores a la precisión de las predicciones?*
Los sensores de los contenedores tienen un ratio de envío configurable que determina cada cuanto tiempo toman nuevas muestras. Además, dependen de la infraestructura de la red para transferir datos a la nube. Queremos hallar la frecuencia de muestreo óptima teniendo en cuenta posibles errores de conexión.

RQ3. *¿Cómo mejora nuestro sistema el uso de rutas eficientes*

para la recolección de residuos?
Teniendo predicciones fiables tras contestar a la anterior pregunta, el siguiente paso es crear rutas acordes para la recogida de residuos. Aquí exploraremos las ventajas de generarlas mediante un algoritmo evolutivo frente a dejar la decisión en manos de personas expertas.

Las contribuciones principales de nuestro trabajo son las siguientes:

- Un sistema inteligente que predice de manera precisa los valores de llenado futuros de contenedores mediante técnicas de *Deep Learning* y genera rutas eficientes para la recogida de residuos utilizando un algoritmo evolutivo.
- Una aplicación web para monitorizar y administrar el sistema en su conjunto.

El resto de artículo está organizado de la siguiente manera. La Sección II describe los principales componentes de nuestro sistema tanto en hardware (contenedores compactadores sensorizados) como en software (predicción de generación de residuos y generación de rutas). La Sección III contiene la descripción de un caso de estudio así como los resultados de los tres experimentos realizados sobre el mismo. Finalmente, la Sección IV concluye el artículo y expone posible trabajo futuro.

II. BIN-CT

BIN-CT es un sistema de IA predictiva para la gestión eficiente de recogida de residuos que se divide en dos partes principales: (i) los contenedores sensorizados con mecanismos de compactación y (ii) un sistema software ad hoc que procesa los datos recogidos por los contenedores, predice niveles de llenado y calcula rutas eficientes para su recogida.

II-A. Contenedor compactador sensorizado

Los contenedores han sido diseñados y desarrollados con la ayuda de las empresas Eurosoterrados, CETEM, y ACTECO. Tienen forma cúbica y están equipados con sensores que emiten datos periódicamente mediante el protocolo Sigfox para ser posteriormente almacenadas en una base de datos. Esta información incluye nivel de llenado (porcentaje), temperatura (Celsius), humedad (porcentaje), localización (lat/lon) y nivel de batería (porcentaje). Adicionalmente, los contenedores están equipados también con una prensa vertical. El objetivo de este mecanismo es incrementar la capacidad efectiva de los mismos. Esta prensa podría representar un peligro para los usuarios del contenedor, así que se han tomado medidas de seguridad para minimizar los riesgos. En primer lugar, la prensa solo puede ser activada manualmente con un botón en la superficie del contenedor. En segundo lugar, un conjunto de sensores vigilan la boca del contenedor y detienen la prensa si detectan movimiento alrededor de la misma.

II-B. Sistema software

El software es una parte crucial de BIN-CT. Nuestro sistema está compuesto por dos partes principales:
1) Un *core* interno que encapsula las principales funcionalidades en cuanto a procesado de datos. Está desarrollado en Java y dividido en tres módulos. El primero procesa la información generada por los sensores para generar predicciones de llenado diarias. El segundo utiliza tanto estas predicciones como los niveles de llenado para seleccionar los contenedores que serán recogidos cada día. El último se encarga de generar las rutas óptimas en función de los contenedores seleccionados.
2) Una interfaz web para el manejo del sistema diseñada de manera simple, completa e intuitiva. La aplicación ha sido desarrollada en Angular, utilizando TypeScript como principal lenguaje de programación. Esta web tiene dos partes diferenciadas: el *backend* que interpreta la información de las bases de datos y el *core* interno, y un *frontend* que define la capa de presentación y sirve de puente entre el usuario y la información.

El *core* de BIN-CT calcula niveles de llenado diarios, es decir, cuantos residuos han sido depositados en cada contenedor por día. De esta manera, aislamos la cantidad de residuos generados de eventos como compactaciones y vaciados, que afectan directamente al nivel del contenedor. Nos referimos al dato de llenado diario como histórico. Los datos históricos sirven a nuestro algoritmo predictivo para conseguir pronósticos a corto plazo.

II-B1. Predicciones de generación de residuos: Para las predicciones de nivel de llenado de cada contenedor, BIN-CT utiliza una Red Neuronal Recurrente (RNN), un tipo específico de Red Neuronal Artificial (ANN). Al igual que otras ANNs, este tipo de red está compuesta de múltiples capas ocultas entre las capas de entrada y salida. Las RNNs incorporan conexiones de alimentación y retroalimentación entre capas para identificar dependencias a largo plazo en los datos de entrada. El uso de RNNs nos permite generar predicciones para un conjunto de contenedores utilizando una única RNN en vez de crear y entrenar una para cada contenedor. Esto supone una ventaja puesto que se necesita menos cantidad de datos gracias a la explotación de las similitudes entre contenedores.

Aplicamos también aprendizaje supervisado, que consiste en un proceso iterativo que requiere de un conjunto de datos de entrenamiento (N pares de entrada-salida). Como estamos tratando con predicciones de nivel de llenado, los datos de entrada son los niveles históricos de llenado de los contenedores, y los de salida los niveles para el siguiente día. Es decir, la ANN produce una salida (cantidad de residuo generada) que es comparada con el dato real utilizando una función de error. Tras esto, se aplica un proceso para reducir este error actualizando la red hasta que se alcanza cierto criterio de parada [10].

Puesto que entrenar una RNN es costoso (en términos de capacidad computacional) y el número de arquitecturas de RNN es infinita (o extremadamente grande si imponemos restricciones en el número de capas ocultas), estamos forzados a definir una estrategia de búsqueda inteligente para encontrar una RNN óptima. En este estudio, aplicamos un enfoque de neuroevolución profunda para la generación de predicciones. Nuestro método de optimización determina los siguientes parámetros: el *lookback*, el número de capas ocultas y el número de neuronas por capa oculta. Decidimos entrenar

la red utilizando *backpropagation* y midiendo la *media del error absoluto* (MAE) de los valores predichos respecto a los observados.

La ecuación 1 representa el problema de encontrar una arquitectura óptima como un problema de minimización, donde *N* es el número de pares en el conjunto de datos de prueba (X, Y), z_i e y_i corresponden al valor predicho y al valor verdadero de la *i*-ésima muestra, respectivamente. Nótese que la RNN es alimentada con datos de predicción $\hat{x}$, y que la arquitectura está restringida por *B*, *H* y *L*. Para resolver el problema planteado en la ecuación 1 utilizamos un algoritmo neuroevolutivo profundo basado en la $(1+1)$ *Evolutionary Strategy* (ES) [11] y en el *Adam weights optimizer* [12].

$$\begin{aligned} \text{minimizar} \quad & \textit{fitness} = \frac{1}{N}\sum_{i}^{N} MAE(z_i, y_i) \qquad (1) \\ \text{sujeto a} \quad & B \leq \textit{max\_look\_back} \\ & H \leq \textit{max\_hidden\_layers} \\ & L \leq \textit{max\_neurons\_per\_layer} \\ & \hat{x}_i = \begin{cases} x_0 & \text{if } i = 0 \\ z_{i-1} & \text{if } i > 0 \end{cases} \end{aligned}$$

II-B2. Generación de Rutas: Para poder generar rutas de recogida eficientes utilizamos un Algoritmo Evolutivo (AE) (1+1). El AE utilizado está basado en el principio de ruina y recreación. Se trata de una búsqueda local que combina elementos de *simulated annealing* y algoritmos *threshold-accepting*. Este enfoque es apropiado para problemas complejos con muchas restricciones [13], [14]. Es una metaheurística multipropósito que puede ser utilizada para resolver diferentes problemas clásicos de enrutado de vehículos, pudiendo variar las estrategias de búsqueda en función del problema. El Algoritmo 1 muestra el pseudocódigo del mismo. El algoritmo empieza con una solución inicial aleatoria. Extrae partes de la solución que llevan a un conjunto de tareas que no serán asignadas a un vehículo, obteniendo así la solución parcial. A este proceso se le llama arruinar la solución porque extraemos una parte de una solución. Basándonos en la solución parcial, el algoritmo reintroduce las tareas extraídas llevando a una nueva solución. Este es el proceso de recreación. Si la nueva solución es igual o de mejor calidad que la anterior, se acepta como nueva mejor solución. Estos pasos se repiten hasta alcanzar cierto número de iteraciones.

Algoritmo 1 Pseudocódigo de la generación de rutas.

Entradas: Conjunto de contenedores: (localización, nivel de llenado, predicciones), conjunto de vehículos, matriz de distancias y tiempos, ubicación de la planta de reciclaje
Salidas: Conjunto de listas de contenedores ordenados

```
solucionActual ← solucionAleatoria()
mejorSolucion ← solucionActual
evals ← 0
while evals < totalEvals do
    tareasQuitadas← ruina(solucionActual)
    solucionActual← recreacion(tareaQuitada)
    evaluar(solucionActual)
    if (solucionActual < mejorSolucion) then
        mejorSolucion←solucionActual
    end if
    evals ← evals + 1
end while
return mejorSolucion
```

III. Experimentos

En esta sección realizamos un estudio experimental para validar BIN-CT. Con el objetivo de responder las preguntas de investigación expuestas en la Sección I, dividimos el estudio en tres experimentos diferentes. Primero, tratamos de exponer los beneficios de utilizar BIN-CT respecto a los medios tradicionales de gestión de residuos. En el segundo experimento analizamos el impacto que tiene la frecuencia de medidas obtenidas por los sensores de los contenedores en nuestro sistema. Finalmente, en el tercer experimento, enfatizamos los beneficios de utilizar rutas eficientes en la recogida de residuos. Para la realización de estos experimentos, introduciremos primero el caso de estudio, así como el conjunto de datos utilizados.

III-A. Caso de estudio y conjunto de datos

Nuestro caso de estudio se centra en 80 hoteles y hostales alrededor de Alicante, España, donde simulamos el despliegue de contenedores compactadores sensorizados para residuos plásticos (un contenedor por establecimiento).

El área urbana seleccionada incluye las localidades de Ibi (donde se encuentra la planta de reciclaje) y la ciudad de Alicante (donde e encuentran la mayoría de los establecimientos). El área total de la muestra es de aproximadamente 250 km^2. Para poder realizar los experimentos, utilizamos un conjunto de datos sintético pero generado de manera realista, conteniendo niveles de llenado, compactaciones y vaciados durante un mes 80 contenedores. No disponemos de datos reales para los experimentos debido a que, al ser éste un proyecto piloto, aún no hay 80 contenedores listos para su despliegue. Cuando simulamos la creación de residuos plásticos en los hoteles, hemos considerado patrones que se observan en el mundo real, como son los siguientes:

- Los niveles de llenado están distribuidos aleatoriamente con incrementos sustanciales, en vez de tener un crecimiento constante en el tiempo.
- Los contenedores se separan en dos categorías según su velocidad de llenado: de llenado rápido (10 % de probabilidad de llenarse entre un 0 % y un 8 % cada 15 minutos) y de llenado lento (10 % de probabilidad de llenarse entre un 0 % y un 4 % cada 15 minutos).
- Los niveles de llenado no se incrementan de 2:00 AM a 7:00 AM, puesto que no suele haber actividad en esas horas.
- Los contenedores son vaciados si exceden cierto nivel de llenado (75 %) entre las 8:00 AM y las 9:00 AM, no siendo este umbral un tope a partir del cual no se llenen.
- Cada contenedor permite hasta 3 compactaciones entre recogidas, cada vez que el contenido excede cierto nivel de llenado (80 %). Cada una de estas compactaciones

incrementa, aleatoriamente, la capacidad del contenedor dentro de los rangos 20-30 %, 10-20 %, y 0-10 %, respectivamente.

Para la recogida de residuos disponemos de dos tipos de camiones: pequeños, con capacidad para el contenido de hasta 5 contenedores completos, y grandes, con el doble de capacidad. El coste aproximado es de 50€ y 100€ por hora (camión y conductor), respectivamente.

III-B. Experimento 1. Recogida tradicional vs. BIN-CT

Este experimento consiste en una comparación entre los métodos de recogida de residuos tradicionales y BIN-CT. Del conjunto de datos expuesto anteriormente, hemos escogido para este experimento un subconjunto de 20 contenedores con capacidad y velocidad de llenado similares. Este tamaño de subconjunto fue decidido porque esperábamos poder resaltar cómo las diferencias entre sistemas son claramente visibles incluso a tan pequeña escala. Simulamos una única iteración del plan de recogida durante tres días. Después, analizamos el estado del conjunto para el sistema tradicional y para BIN-CT. El sistema tradicional consistirá en una ruta fija que incluya todos los contenedores, puesto que tienen velocidades de llenado similares, proyectada para el día en el que se suponen llenos.

La Tabla I muestra un resumen del estado de los contenedores tras el tercer día del periodo estudiado. Si miramos los resultados de los contenedores tradicionales (segunda columna) son necesarios dos camiones grandes para la recogida de residuos. Aunque hay cuatro contenedores que podrían ser dejados fuera de ruta por no haber alcanzado el límite para recogida (80 %), deben ser visitados porque no es imposible conocer su estado tanto actual como la previsión para el final del día sin monitorización. Por otra parte, si miramos los resultados de contenedores compactadores sensorizados (tercera columna) solo es necesario un camión pequeño, puesto que solo hay tres contenedores con valores de llenado por encima del límite permitido. Utilizando los importes expuestos anteriormente, este dato por sí mismo supone un ahorro de 150€ por hora utilizando contenedores compactadores sensorizados.

	Tradicional	BIN-CT
Contenedores sobre el límite	16	3
Contenedores bajo el límite	4	17
Contenedores sobre 100 %	7	0
Nivel máximo	53 %	49 %
Nivel mínimo	112 %	84 %
Nivel medio	91,6 %	65,8 %
Camiones necesarios	x2 grandes	x1 pequeño

Tabla I: Comparación entre gestión de residuos tradicional y nuestra propuesta para un plan de recogida en un periodo de tres días.

En cuanto al nivel de llenado, tras tres días los contenedores tradicionales están al 91.6 % de su capacidad, de media. Por contra, nuestros contenedores compactadores sensorizados tienen de media un 65.8 % de su capacidad ocupada. Esto se debe al incremento de capacidad efectiva causado por las compactaciones. En los contenedores compactadores sensorizados se realizaron en este periodo 1.2 compactaciones de media.

Si consideramos un día más (el cuarto), nuestros contenedores compactadores sensorizados aún no alcanzan los mismos valores de llenado que tienen los tradicionales, llegando tan solo a una media de 83.4 % con 2.85 compactaciones. Este incremento en el tiempo entre llenados de 1-2 días respecto a contenedores tradicionales se traduce directamente en una mayor eficiencia del sistema. El aumento de capacidad hace que los contenedores compactadores sensorizados sean vaciados de media unas 2-3 veces menos por mes.

Contestando a la pregunta **RQ1**, un menor número de vaciados a través del tiempo implica menos camiones necesarios para dar un servicio de calidad y, por tanto, un ahorro en costes y una reducción de emisiones contaminantes.

III-C. Experimento 2. Tolerancia a la pérdida de datos.

En este experimento analizaremos el impacto de la reducción en las medidas tomadas y la pérdida de datos en la precisión de las predicciones de nivel de llenado. El objetivo es discernir de qué cantidad de datos podemos prescindir antes de que nuestra interpretación de estos sea errónea.

Para evaluar el comportamiento de nuestro sistema con pérdida de datos, borramos cierto porcentaje de datos del conjunto de datos original de manera aleatoria. Este porcentaje empieza en un 10 % de las muestras totales, y escala proporcionalmente de 10 en 10 para tener muestra de datos desde un 0 % de pérdida hasta un 90 %. Con estos conjuntos de datos se computan predicciones para los siguientes siete días.

La Tabla II muestra la media de las predicciones de nivel de llenado para los diferentes niveles de pérdida aleatoria, tanto para contenedores de llenado lento y rápido por separado como todos juntos. Las dos primeras columnas muestran el porcentaje de datos perdidos y la media de datos por día en cada escenario, respectivamente. El resto de columnas muestran el nivel de predicción de llenado medio y desviación típica, y la diferencia con el valor base (sin pérdida) para cada caso. Los porcentajes bajos de pérdida de datos tienen un impacto casi nulo en la robustez del sistema. Como era esperable, no obstante, porcentajes altos de pérdida incrementan significativamente el error en las predicciones. Respecto a las diferencias entre llenado lento y rápido, es claramente visible que cuanto más rápido se llena un contenedor, mayor es el error producido por la pérdida de información.

Hemos realizado un test de Kruskal-Wallis para muestras independientes sobre nuestros datos de predicción, agrupando por porcentaje de pérdida. Este test es un método no paramétrico que refleja si varios grupos de datos pueden ser considerados parte de una misma distribución o no. Cuando tenemos en cuenta las medias de todos los contenedores, los resultados obtenidos son prometedores. Perdiendo hasta un 40 % de los datos totales las predicciones siguen siendo lo suficientemente precisas como para que no puedan ser consideradas de diferentes distribuciones. Por otra parte, cuando probamos contenedores de llenado lento y rápido por separado, el salto en calidad es considerable. La robustez de

		Llenado lento		Llenado rápido		Todos	
Pérdida	#Muestras	Media	Dif.	Media	Dif.	Media	Dif.
0 %	96	$13,76_{1,68}$	0,00	$26,04_{2,75}$	0,00	$19,89_{6,55}$	0,00
10 %	89	$13,74_{1,66}$	-0,02	$25,64_{2,62}$	-0,39	$19,69_{6,35}$	-0,19
20 %	77	$13,59_{1,62}$	-0,17	$25,25_{2,80}$	-0,79	$19,42_{6,27}$	-0,47
30 %	67	$13,66_{1,67}$	-0,11	$24,93_{2,62}$	-1,10	$19,30_{6,06}$	-0,59
40 %	58	$13,50_{1,68}$	-0,26	$24,52_{2,64}$	-1,52	$19,01_{5,94}$	-0,87
50 %	48	$13,38_{1,69}$	-0,38	$23,89_{2,54}$	-2,15	$18,64_{5,68}$	-1,25
60 %	38	$13,44_{1,70}$	-0,33	$23,40_{2,30}$	-2,64	$18,42_{5,38}$	-1,47
70 %	29	$13,49_{1,81}$	-0,27	$22,39_{2,19}$	-3,64	$17,94_{4,88}$	-1,94
80 %	19	$12,58_{1,53}$	-1,19	$20,68_{2,19}$	-5,35	$16,63_{4,48}$	-3,26
90 %	10	$12,32_{1,89}$	-1,45	$17,86_{2,34}$	-8,18	$15,09_{3,49}$	-4,79

Tabla II: Media de las predicciones de nivel de llenado para las diferentes muestras. Los subíndices muestran desviaciones estándar.

las predicciones en contenedores de llenado lento se mantiene incluso perdiendo hasta un 70 % de los datos de entrada, mientras que los de llenado rápido experimentan un mayor descenso de calidad, manteniendo la fidelidad solo hasta el 20 % de datos perdidos.

La Figura 1 muestra el error medio de las predicciones para contenedores de llenado lento y rápido. En ella se puede ver que el error aumenta considerablemente para los valores más grandes de pérdida de datos aleatoria.

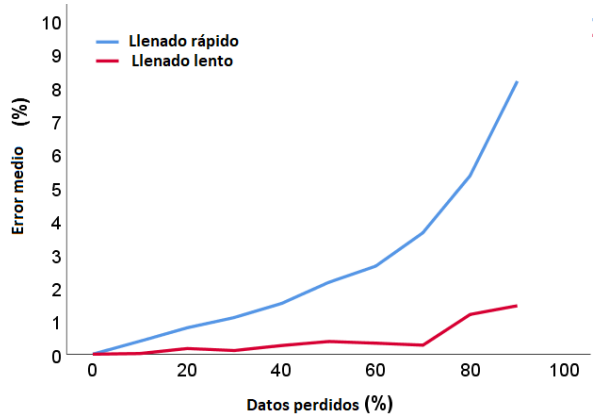


Figura 1: Media de errores en las predicciones para contenedores de llenado lento y rápido.

Con este experimento se remarca lo resistente que es nuestro sistema en general a la pérdida de datos, abriendo así la puerta a una reducción en el ratio de envío de datos de nuestros sensores. Esto podría ayudar a ahorrar tanto energía como costes relativos al envío de datos, así como dar una respuesta a la **RQ2**. Los resultados de este experimento indican que podemos reducir la tasa de envío de información de nuestros sensores a casi la mitad de la original sin que esto repercuta en la calidad del servicio prestado.

III-D. Experimento 3. Expertos vs. IA para enrutado

En este experimento comparamos la eficiencia del uso de un algoritmo evolutivo para la resolución de un problema de enrutado respecto a la solución diseñada por personas con experiencia en el campo. Con esto queremos cuantificar los beneficios de generar rutas de recogida de residuos mediante IA.

Para realizar este experimento, hemos escogido 30 contenedores del conjunto de contenedores inicial. Los hemos seleccionado con el objetivo de que estuvieran distribuidos por toda la ciudad, teniendo tanto contenedores en zonas aisladas como pequeños conjuntos próximos entre sí. De estos contenedores, 11 deben ser vaciados en ese mismo día mientras que los 19 restantes pueden ser vaciados o no. Hemos escogido un número relativamente pequeño de contenedores para hacer más viable el que un humano pudiera ser competitivo frente a un algoritmo.

Hemos entregado un mapa interactivo con la localización de los contenedores a siete personas con experiencia resolviendo problemas de enrutado de vehículos. Hay dos camiones disponibles para la recogida de residuos en esta versión del problema, uno pequeño y uno grande. Además, el punto de comienzo y final de ruta para ambos camiones es la planta de reciclaje, situada a unos 40 kilómetros al norte de la ciudad. Hemos pedido a nuestros expertos que encuentren una ruta eficiente para recoger todos los contenedores posibles incluyendo todos los obligatorios y los opcionales posibles, minimizando tiempo y distancia recorrida. A la vez, ejecutamos nuestro algoritmo para que genere su propia ruta eficiente.

La Tabla III muestra la distancia (en kilómetros), el tiempo (en horas y minutos) y las diferencias (en porcentaje) de la solución de cada experto respecto al algoritmo evolutivo descrito en la Sección II-B2, en términos de tiempo y distancia recorrida. Las rutas de los expertos son bastante similares en términos de eficiencia: todas están en un rango de 25 kilómetros y con no más de media hora de diferencia entre la más lenta y la más rápida. Hemos encontrado ciertos patrones en las elecciones de nuestros expertos. Cinco de los siete expertos (71 %) enviaron el camión pequeño a la parte nordeste de la ciudad al principio de su ruta. Esta estrategia ha resultado ser, no obstante, poco eficiente en términos de tiempo, ya que los expertos 1 y 6 tienen un ratio distancia/tiempo mucho mejor en sus rutas, siendo los que no tomaron esa decisión. Esta ineficiencia es debida probablemente al hecho de que las carreteras que van de la planta de reciclaje a esa zona de la ciudad tengan una mayor densidad de tráfico, mayor número de semáforos o menor límite de velocidad, entre otros factores.

La distancia y el tiempo de la ruta proporcionada por el algoritmo evolutivo son 175,40km y 3 h 03 m respectivamente. Esta ruta reduce distancia y tiempo en un 32,57 % y 57,14 % respecto a la mejor ruta propuesta por los expertos. Esta mejora se traduce en una reducción de costes y emisiones de gases de efecto invernadero. Además, teniendo en cuenta los precios de ambos tipos de camiones, se ahorrarían más de 150€ solo en ese día utilizando la ruta generada por la IA en vez de cualquiera de las generadas por expertos. Este ahorro acumulativo conforme el tiempo pase es precisamente la eficiencia económica que esperábamos alcanzar cuando propusimos la **RQ3**.

En este experimento, la técnica de generación de rutas propuesta es capaz de generar una ruta mucho más eficiente

	Rutas		Diferencias con AE	
Nombre	Dist. (km)	Tiempo	Dist. (%)	Tiempo (%)
Algoritmo	175,40	3 h 03 min	-	-
Experto 1	232,80	4 h 48 min	32,57	57,14
Experto 2	245,40	4 h 53 min	40,00	60,05
Experto 3	232,80	5 h 26 min	32,57	77,81
Experto 4	242,50	5 h 18 min	38,29	73,56
Experto 5	257,70	5 h 04 min	47,43	65,87
Experto 6	257,70	5 h 13 min	47,42	70,78
Experto 7	242,20	4 h 55 min	38,30	61,40

Tabla III: Comparación entre soluciones de expertos y algoritmo evolutivo para el problema de enrutado de vehículos propuesto en el tercer experimento.

que los expertos incluso para un problema a pequeña escala como el aquí presentado.

Un simple incremento en el número de contenedores, el número de camiones o incluso tratar con llenados en tiempo real hacen el problema intratable para un ser humano. En esos casos, los beneficios de usar un algoritmo de IA para la generación de rutas serían incluso más reseñables.

IV. Conclusiones y líneas de trabajo futuras

En este artículo hemos propuesto BIN-CT, un sistema de IA predictivo para la recogida eficiente de residuos. Con la ayuda de tecnologías IoT, hemos sido capaces de diseñar, construir y probar varias unidades de un novedoso contenedor equipado con diferentes sensores que emiten datos periódicamente. Estos contenedores también incluyen un mecanismo de compactación que ayuda a aumentar su capacidad efectiva. Los datos emitidos por nuestros contenedores compactadores sensorizados son procesados y utilizados para realizar predicciones y calcular rutas de recogida eficientes para llevar los residuos plásticos a una planta de reciclado.

Nuestros contenedores compactadores sensorizados presentan ventajas tanto en el lado económico como en el ecológico. El mecanismo de compactación incrementa la capacidad efectiva del contenedor y, por tanto, aumenta el tiempo entre vaciados. En el Experimento 1, se pudo observar que el mecanismo de compactación aumentó el tiempo necesario para recoger un contenedor en 1-2 días, reduciendo el número de viajes necesarios en el tiempo y reduciendo así costes y emisiones contaminantes.

Basándonos en los resultados del Experimento 2, podemos decir que el algoritmo de RNN que predice el nivel de llenado es muy tolerante a la pérdida de datos. Concretamente, el error medio para todos los contenedores en una semana de predicciones es de menos del 5 %, incluso al perder el 90 % de las medidas diarias de los sensores. Una vez el sistema ha decidido el conjunto de contenedores que deben ser recogidos en un día, el algoritmo de enrutado de vehículos provee una significativamente mejor solución que la que pueden diseñar humanos expertos. Como reflejan los resultados del Experimento 3, la solución del algoritmo evolutivo es un 32.57 % mejor en distancia recorrida y un 57.14 % mejor en tiempo necesario respecto a la mejor solución de los expertos. El uso de rutas eficientes también se traduce en una reducción de costes y contaminación.

Planeamos extender BIN-CT para también incluir otros tipos de residuos reciclables como cartón, vidrio o metal. Para que esto sea posible sería necesario realizar más experimentos, puesto que estos materiales se comportan de una manera diferente cuando se someten a presión, pudiendo necesitar sistemas diferentes para aumentar la capacidad efectiva de los contenedores. Cuanto más tipos de residuos maneje el sistema, más cerca estaremos de alcanzar la sostenibilidad total en nuestro área de influencia.

Agradecimientos

Esta investigación ha sido financiada parcialmente por los proyectos Eco-IoT (RTC-2017-6714-5), TAILOR (GA No 952215) y por la Universidad de Málaga, Campus Internacional de Excelencia Andalucía TECH. Además, Rubén Saborido es beneficiario de la beca Juan de la Cierva (referencia FJC2018-038537-I) y Javier Ferrer es beneficiario de una beca post-doctoral (DOC/00488) de la Junta de Andalucía.

Referencias

[1] W. Leal Filho, U. Saari, M. Fedoruk, A. Iital, H. Moora, M. Klöga, and V. Voronova, "An overview of the problems posed by plastic products and the role of extended producer responsibility in europe," *Journal of cleaner production*, vol. 214, pp. 550–558, 2019.

[2] I. Manisalidis, E. Stavropoulou, A. Stavropoulos, and E. Bezirtzoglou, "Environmental and health impacts of air pollution: a review," *Frontiers in public health*, vol. 8, 2020.

[3] J. Korhonen, A. Honkasalo, and J. Seppälä, "Circular economy: the concept and its limitations," *Ecological economics*, vol. 143, pp. 37–46, 2018.

[4] S. Oduro-Kwarteng, *Private Sector Involvement in Urban Solid Waste Collection: UNESCO-IHE PhD Thesis*. CRC Press, 2011.

[5] H. Niska and A. Serkkola, "Data analytics approach to create waste generation profiles for waste management and collection," *Waste Management*, vol. 77, pp. 477–485, 2018.

[6] N. E. Johnson, O. Ianiuk, D. Cazap, L. Liu, D. Starobin, G. Dobler, and M. Ghandehari, "Patterns of waste generation: A gradient boosting model for short-term waste prediction in new york city," *Waste management*, vol. 62, pp. 3–11, 2017.

[7] E. Shamshiry, M. Mokhtar, A.-M. Abdulai, I. Komoo, and N. Yahaya, "Combining artificial neural network-genetic algorithm and response surface method to predict waste generation and optimize cost of solid waste collection and transportation process in langkawi island, malaysia," *Malaysian Journal of Science*, vol. 33, no. 2, pp. 118–140, 2014.

[8] C. J. Baby, H. Singh, A. Srivastava, R. Dhawan, and P. Mahalakshmi, "Smart bin: An intelligent waste alert and prediction system using machine learning approach," in *2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*. IEEE, 2017, pp. 771–774.

[9] Y. Zhao, S. Yao, S. Li, S. Hu, H. Shao, and T. F. Abdelzaher, "Vibebin: A vibration-based waste bin level detection system," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 1, no. 3, pp. 1–22, 2017.

[10] S. Haykin, *Neural networks and learning machines*. Pearson, 2009, vol. 3.

[11] T. Back, *Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms*. Oxford university press, 1995.

[12] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[13] A. Misevicius, "Genetic algorithm hybridized with ruin and recreate procedure: Application to the quadratic assignment problem," in *Knowledge-Based Systems*, vol. 16, no. 5-6 SPEC., 2003, pp. 261–268.

[14] G. Schrimpf, J. Schneider, H. Stamm-Wilbrandt, and G. Dueck, "Record Breaking Optimization Results Using the Ruin and Recreate Principle," *Journal of Computational Physics*, vol. 159, no. 2, pp. 139–171, 2000.

Sistema de Recomendación con Explicaciones Basadas en Texto

Pablo Pérez-Núñez
Centro de Inteligencia Artificial
Universidad de Oviedo
Gijón, España
pabloperez@uniovi.es

Antonio Bahamonde
Centro de Intel. Artificial
Universidad de Oviedo
Gijón, España
abahamonde@uniovi.es

Oscar Luaces
Centro de Intel. Artificial
Universidad de Oviedo
Gijón, España
oluaces@uniovi.es

Jorge Díez
Centro de Inteligencia Artificial
Universidad de Oviedo
Gijón, España
jdiez@uniovi.es

***Resumen*—A los usuarios de los sistemas de recomendación cada vez les interesa más que las recomendaciones vengan acompañadas de explicaciones que les permitan comprender las razones por las que un determinado producto es recomendado. Muchos sistemas ofrecen justificaciones basadas en la cercanía entre productos o usuarios en lugar de explicaciones extraídas del modelo aprendido por el sistema. En este artículo presentamos un sistema de recomendación que ofrece explicaciones extraídas de los valores aprendidos para una matriz que relaciona las palabras relevantes de las reseñas con los productos. Además, al mismo tiempo aprende la valoración asociada a cada reseña, de manera que el sistema será capaz de recomendar una serie de productos, explicar en qué basa cada una de sus recomendaciones y anticiparse a la valoración que daría el usuario a esos productos si se decide por alguno de ellos.**

***Index Terms*—Sistemas de Recomendación, Inteligencia Artificial Explicable, Procesamiento del Lenguaje Natural, Multitarea**

I. Introducción

Cuando accedemos a cualquier gestor de contenidos, como por ejemplo una tienda online, la disposición de la información que visualizamos suele tener como propósito incentivar nuestro consumo. En una tienda online se pretende que compremos productos, en un periódico digital que leamos unas noticias y en un servicio de streaming que veamos o escuchemos determinados contenidos. Se ponen a la vista del usuario aquellos productos que se quieren promocionar o aquellos que se piensa que le pueden agradar y que por tanto tienen mayor probabilidad de ser consumidos por el usuario. Cada consumidor tiene sus propios gustos y para poder identificar aquellos productos que agradan a cada uno de los usuarios se han desarrollado unos algoritmos conocidos como Sistemas de Recomendación (SR) ([1], [2]). Estos sistemas analizan las interacciones de todos los usuarios con todos los productos y generan modelos capaces de hacer recomendaciones personalizadas para cada usuario.

Estas recomendaciones que reciben los usuarios suelen ser acertadas ya que se ha comprobado que, efectivamente, incrementan las ventas de una determinada tienda o el consumo de los contenidos ofertados en una plataforma [3]. Por tanto, desde el punto de vista de la empresa, el SR funciona bien, pues hace incrementar sus beneficios. Desde el punto de vista de los consumidores parece que el SR también es efectivo, ya que si un usuario consume más es bastante probable que se deba a que se le han ofrecido productos de su agrado.

Sin embargo, en los últimos años ha crecido la desconfianza hacia los algoritmos que tratan de ayudar a los humanos a tomar decisiones. Esta desconfianza nace de la posibilidad de que estos algoritmos tengan un cierto sesgo [4] y traten de orientarnos a tomar decisiones que no sean las mejores para nuestros intereses. Para evitar estas suspicacias nace una disciplina que en los últimos años ha cobrado especial relevancia, la *Inteligencia Artificial Explicable* (XAI) [5]. Lo que pretende la XAI es explicar las decisiones de los algoritmos para que los humanos comprendan el porqué de tales decisiones y de esta manera recuperar la confianza.

Evidentemente, la XAI se está aplicando también en el campo de los SR [6], donde ofrecer explicaciones a los usuarios puede redundar en varios beneficios. Las explicaciones ofrecen i) *transparencia*, ya que se le está dando al usuario la oportunidad de comprender cómo funciona el sistema; ii) *que el sistema se pueda escrutar*, puesto que al conocer su funcionamiento, cuando se detectan malas recomendaciones se puede analizar a qué son debidas e intervenir para mejorar el modelo; iii) *confianza*, puesto que el usuario recibe una explicación de la recomendación; iv) *modelos convincentes*, ya que la explicación puede incidir sobre aspectos del producto en los que realmente estamos interesados; v) *efectividad y eficiencia*, como los productos recomendados son adecuados y se explica el porqué de su recomendación, el consumidor realiza rápidas y buenas elecciones; y vi) *safisfacción*, puesto que el usuario finaliza habiendo elegido un producto adecuado y no ha perdido tiempo en la elección [7].

Por otra parte, los SR suelen aprender a partir de valoraciones que realizan los usuarios de los productos que ya han comprado o disfrutado y generan modelos capaces de hacer recomendaciones personalizadas en base al histórico de interacciones mostradas por cada usuario. Sin embargo, en algunos contextos existe una escasez de datos en cuanto al número de productos evaluados por cada usuario individualmente, aunque los productos reciban un número elevado de valoraciones. Un ejemplo que describe este contexto son los *cruceros*, donde

Este trabajo ha sido financiado por el proyecto PID2019-109238GB-C21 del Ministerio de Ciencia e Innovación de España. La colaboración de Pablo Pérez-Núñez ha sido financiada por el Gobierno del Principado de Asturias mediante el programa de becas predoctorales *Severo Ochoa* (ref. BP19-012)

podemos contar con muchas reseñas de cada crucero pero lo normal es que no haya un número significativo de usuarios con reseñas de muchos cruceros diferentes. Esta situación imposibilita la elaboración de SR en los que intervenga el usuario directamente.

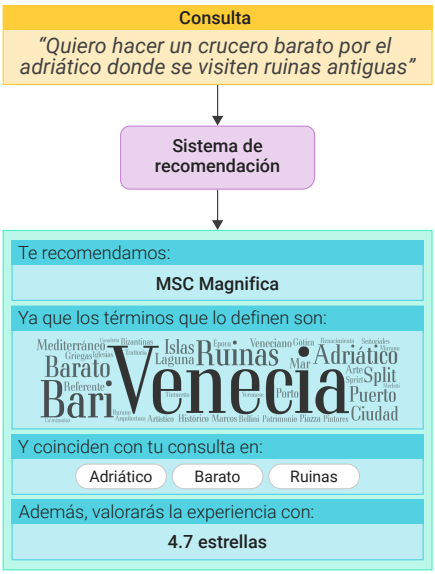


Figura 1. Continuando con el ejemplo de los cruceros, el sistema recibirá como entrada una consulta en texto libre y generará una recomendación con uno o varios cruceros explicando el porqué de la recomendación. Además mostrará la valoración estimada de la experiencia si se disfruta de alguno de los cruceros recomendados

Para sortear esta dificultad, en este artículo vamos a realizar un SR basado principalmente en las reseñas escritas por los usuarios. Este recomendador responderá a consultas realizadas por los usuarios que indicarán sus necesidades mediante un texto libre en lenguaje natural, como puede verse en la Figura 1. El SR no necesitará tener un conocimiento previo del usuario, con lo que también se presenta como solución para el problema del *arranque en frío* (cold-start) [8].

Además, este SR será capaz de ofrecer a los usuarios recomendaciones con explicaciones basadas en texto a partir de las reseñas que han escrito otros usuarios de los productos o contenidos que se están analizado. Para ello, se extraerán de las reseñas aquellos términos en los que los usuarios hacen especial hincapié y que además son descriptivos. Utilizando estos términos en una *bolsa de palabras* (bag of words, BoW) junto con un análisis semántico de las palabras (usando *word2vec* [9]) y de las reseñas (mediante redes *long short-term memory* (LSTM) [10]) desarrollaremos un sistema capaz de recomendar un conjunto de productos explicando por qué se recomienda cada uno de ellos y además mostrando una valoración del grado de satisfacción que tendrá el consumidor si se decide por alguno de los productos recomendados.

En las siguientes secciones comentaremos trabajos relacionados con la idea presentada en este artículo para, posteriormente, detallar el método propuesto. Luego describiremos el conjunto de datos utilizado y comentaremos los resultados obtenidos para finalizar exponiendo las conclusiones alcanzadas.

II. Trabajo Relacionado

A la hora de ofrecer explicaciones en un SR, suele haber una separación clara entre los sistemas que tratan de mostrar cómo se ha elaborado la recomendación y aquellos que se limitan a justificarla [11]. Los primeros son más transparentes y permiten al usuario del SR comprender las causas por las que un producto se le ha recomendado mientras que los segundos suelen apoyarse en relaciones de vecindad entre usuarios y productos para ofrecer la justificación. Si un sistema es transparente y ofrece la posibilidad de ser analizado críticamente por el usuario, entonces estamos ante un sistema *escrutable* [12] que posibilita la mejora del mismo y permite un mejor uso por parte de los usuarios.

En [6] se recoge una extensa bibliografía sobre explicabilidad en los SR, aunque no hay muchos trabajos desarrollados que utilicen el texto de las reseñas. Uno de los primeros trabajos en incorporar el análisis de las reseñas de los usuarios para mejorar las recomendaciones utilizando *aprendizaje profundo* es el de Almahari *et al.* [13] en 2015, donde se utilizan redes neuronales recurrentes, aunque no trata de ofrecer explicaciones. Estas mismas redes se utilizan en el trabajo de Costa *et al.* [14] donde los esfuerzos se enfocan hacia la generación de oraciones explicativas (razonablemente bien formadas) acompañadas de la opinión que el usuario tendría del producto, aunque Bartoli *et al.* [15] alertan del riesgo que supone la generación automática de reseñas puesto que pueden manipular la opinión de los usuarios. Nuestro método, a diferencia del de Costa *et al.*, no pretende generar oraciones (que a veces pueden resultar un tanto artificiales), sino que busca realizar un análisis de la consulta del usuario que sirva como explicación de la recomendación.

Con anterioridad a la irrupción del aprendizaje profundo en el procesamiento del lenguaje natural, el método para la codificación de textos más efectivo y utilizado era el uso de BoW. Este método consiste en determinar un conjunto de términos o palabras que se utilizará para la representación de un texto mediante un vector en el que cada posición se asocia a un término, indicando si éste aparece o no en el texto. Esta manera de representar los textos se ha perfeccionado utilizando técnicas como el TF-IDF o el uso de bigramas, por lo que sigue siendo un método bastante utilizado y que obtiene resultados comparables a los de otros métodos más sofisticados [16]. Es más, cuando se utilizan bigramas como términos en el BoW, en algunas aplicaciones no se aprecia diferencia respecto al uso de *embeddings* de las palabras [17].

III. Método propuesto

Comenzamos presentando el contexto en el que se sitúa el desarrollo del SR. Disponemos de un conjunto de usuarios $\mathcal{U}$ y un conjunto de productos $\mathcal{P}$. Sabemos también que un usuario $u \in \mathcal{U}$ ha disfrutado de uno o varios productos y además ha mostrado su grado de satisfacción mediante una valoración v del producto $p \in \mathcal{P}$ junto con una reseña r con la que

explica por qué ha dado esa valoración. Así, disponemos de un conjunto de datos donde se han almacenado esas interacciones:

$$\mathcal{D} = \{(u, p, [v, r])\}\,. \tag{1}$$

Además, suponemos que estamos en un contexto en el que la mayoría de usuarios no ha generado un histórico de interacciones suficiente como para efectuar recomendaciones basadas en él. Habitualmente presentarán una o dos interacciones únicamente.

En este contexto, se plantea el desarrollo de un SR capaz de realizar recomendaciones a partir de consultas basadas en texto libre y donde no se necesita un histórico de interacciones del usuario. Por tanto, nuestro SR va a tratar de encontrar las relaciones que haya entre las reseñas que escriben los usuarios, la valoración asociada a cada reseña y el producto del que trata.

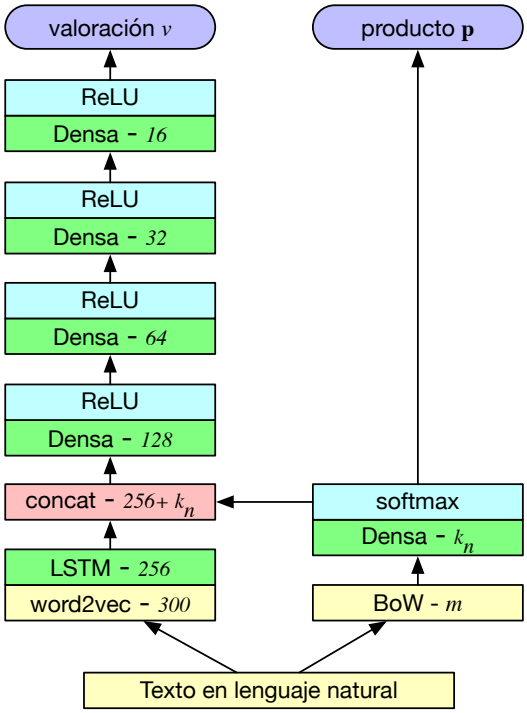


Figura 2. Arquitectura del sistema propuesto.

Así pues, vamos a plantearnos dos objetivos durante el aprendizaje de nuestro sistema. Por un lado, pretendemos extraer del texto de la reseña la información suficiente para ser capaces de *predecir la valoración* a la que ha dado lugar esa reseña. Por otro lado, queremos *identificar el producto* al que se refiere la reseña utilizando únicamente el texto. Además, imponemos a nuestro sistema una restricción adicional, y es que queremos *ofrecer explicaciones* de las recomendaciones. Pretendemos también que esas explicaciones se puedan extraer del modelo aprendido haciéndolo un poco más transparente.

Para cumplir con la restricción de poder ofrecer una explicación que se pueda extraer de los propios parámetros que va a aprender nuestro sistema, sería interesante conocer aquellas palabras que son importantes para definir cada producto. Esta información ya la tenemos disponible, aunque no directamente, puesto que contamos con las reseñas de los usuarios que ya han evaluado los productos y, por tanto, tenemos acceso a aquellos aspectos que los usuarios han utilizado para mostrar su agrado o desagrado con los productos.

Así, el primer paso que tendremos que dar, será identificar un grupo de palabras o términos que tengan importancia en el sector de productos que se esté analizando. Esto es clave para poder obtener posteriormente buenas explicaciones. Una manera sencilla de obtener estas palabras es eliminar las *stop words* y quedarnos con las más frecuentes. Después, podemos seleccionar los m términos más adecuados para describir los productos y ofrecer explicaciones que, normalmente, serán términos relacionados con el contexto de los productos y términos que reflejen sentimientos positivos o negativos.

Utilizando este conjunto de términos, vamos a aprender la importancia de cada uno de ellos para cada uno de los productos, y esto lo vamos a abordar planteando un problema de clasificación donde, a partir de las reseñas codificadas mediante BoW de los términos seleccionados, trataremos de identificar el producto al que se refiere cada reseña. En la parte derecha de la Figura 2 se muestra cómo a partir del texto de la reseña, se obtiene la codificación en BoW de los términos seleccionados que se conecta con una capa de dimensión k_n, que debe coincidir con el número de productos evaluados. Posteriormente se aplica la función de activación *softmax* para obtener la probabilidad con la que esa reseña se refiere a cada uno de los productos:

$$probabilidad = \mathrm{softmax}(\mathbf{W}\mathbf{t} + \mathbf{b}), \tag{2}$$

donde $\mathbf{t}$ representa la codificación BoW del texto de la reseña, $\mathbf{b}$ el vector de términos independientes y $\mathbf{W}$ es la matriz de parámetros que nos lleva de la codificación BoW a la salida y que tendrá dimensión $k_n \times m$. Esta matriz guarda en cada fila la importancia de cada término para un producto determinado y en cada columna lo relevante que es un determinado término para cada uno de los productos. Los parámetros de esta matriz no están acotados y, por tanto, podremos utilizarlos únicamente de manera relativa. Así pues, esta matriz $\mathbf{W}$ nos permitirá conocer qué términos hacen incrementar la probabilidad de que la reseña pertenezca a un determinado producto (signo) y además su incidencia (magnitud). Esto será muy útil para ofrecerle al usuario explicaciones que muestren cómo se elabora la recomendación (transparencia).

Por tanto, utilizando el método descrito hasta el momento, nuestro SR ya es capaz de ordenar los productos a partir de una consulta en forma de texto libre, pudiendo recomendar al usuario el que más probabilidad tenga o mostrarle los 5 con más probabilidad y que el usuario haga la elección final. Además, queremos indicarle al usuario cuál será la valoración que le dará al producto si decide hacer caso de la recomendación. Para ello, vamos a utilizar nuevamente los textos, pero esta vez, como se ve en la parte izquierda de la Figura 2, utilizaremos la codificación *word2vec* de las palabras. Esta codificación puede obtenerse mediante un aprendizaje de la representación de las palabras utilizando únicamente textos relacionados con

los productos en cuestión o, también, se puede descargar una de las múltiples representaciones disponibles en Internet[1] (Caselles-Dupre *et al.* [18] muestran que las codificaciones word2vec obtenidas a partir de los textos de la tarea de recomendación que se esté abordando dan lugar a mejores recomendaciones que si se utilizan codificaciones word2vec genéricas). Posteriormente, se utilizará una capa LSTM para procesar cada texto y generar un vector de dimensión 256 que se concatenará con las probabilidades obtenidas por la otra parte de la red. A partir de ese momento se enlazarán varias capas completamente conectadas y con función de activación *ReLU* para llegar finalmente a la valoración.

Al presentarse dos objetivos de optimización, durante el aprendizaje optimizaremos la suma de las funciones de pérdida asociadas a cada uno de ellos. Por tanto, la función de pérdida global será:

$$\mathcal{L}([v, \mathbf{p}], [\hat{v}, \hat{\mathbf{p}}]) = \mathcal{L}_v(v, \hat{v}) + \mathcal{L}_p(\mathbf{p}, \hat{\mathbf{p}}), \quad (3)$$

donde $\mathcal{L}_p$ representa la función de pérdida asociada a la multi-clasificación para la predicción del producto y $\mathcal{L}_v$ la asociada a la predicción de la valoración que dependerá de cómo sea ésta: si es una valoración numérica se deberá utilizar una función de pérdida asociada a la regresión y si es una valoración del tipo *me gusta / no me gusta* entonces se deberá emplear una función de pérdida específica para clasificación binaria.

IV. Resultados

Para analizar el rendimiento del método propuesto vamos a utilizar un conjunto de datos descargado de TripAdvisor[2]. En esta sección describiremos cómo es el conjunto de datos y comentaremos detalles de implementación necesarios para poder aplicar nuestro método. Posteriormente analizaremos la calidad de las recomendaciones dadas por nuestro método y finalizaremos mostrando cómo se pueden ofrecer explicaciones basadas en la matriz $\mathbf{W}$ de la ecuación (2).

IV-A. Conjunto de Datos Empleado

Hemos descargado de la web de TripAdvisor todas las reseñas escritas en caokstellano por los usuarios de los restaurantes de Gijón. Asimismo, también hemos almacenado la valoración asociada a cada una de las reseñas (que varía entre 1 y 5 estrellas). Por tanto, tenemos disponible un conjunto de ejemplos como el descrito en (1).

Filtramos los datos descargados, quedándonos únicamente con las reseñas de aquellos restaurantes que tienen al menos 100 valoraciones, obteniendo así 39870 ejemplos como los descritos en (1), que se corresponden con 148 restaurantes de Gijón [3]. Se tienen reseñas de 21383 usuarios diferentes que han valorado una media de 1.8 restaurantes cada uno, con lo que nos encontramos ante un problema en el que no se tiene un histórico extenso de los usuarios, tal como se explicaba en la Introducción.

[1] https://github.com/dccuchile/spanish-word-embeddings

[2] https://www.tripadvisor.es

[3] Este conjunto se encuentra disponible para su descarga en https://github.com/pablo-pnunez/TAVtext/blob/CAEPIA/data/raw/gijon.pkl?raw=true

Con el fin de comprobar el rendimiento del sistema propuesto, separamos un 10 % de los ejemplos para test. El 90 % restante se utilizó para obtener el modelo, reservando de nuevo un 10 % de ejemplos para utilizar durante el entrenamiento como conjunto de validación, con el fin de elegir los mejores hiperparámetros.

IV-B. Detalles de Implementación

Para la implementación de la red propuesta en la Figura 2 hemos realizado un programa en *Python* utilizado las librerías TensorFlow, nltk y sklearn para la división del texto. El algoritmo de optimización empleado ha sido Adam [19] y se ha efectuado una regularización mediante el método de la *parada temprana*. La arquitectura propuesta en la Figura 2 requiere definir ciertos hiperparámetros para adaptarla al problema que estamos tratando. Siguiendo el método propuesto en la Sección III, hemos seleccionado $m = 300$ términos que se consideran relevantes para las explicaciones en este contexto. Por otro lado, k_n debe coincidir con el número de productos, en este caso restaurantes, que es 148.

Se utilizó el algoritmo word2vec para aprender de los textos de las reseñas una codificación de las palabras que intervienen en un espacio de 300 dimensiones, utilizando para ello una ventana de tamaño 5.

Otros hiperparámetros como el ratio de aprendizaje o el tamaño del lote fueron elegidos mediante pruebas sobre el conjunto de validación extraído del conjunto de entrenamiento.

Por otra parte, en este conjunto las valoraciones de los usuarios vienen dadas de manera numérica, con lo que la función de pérdida $\mathcal{L}_v$ de la ecuación (3) será el *error cuadrático medio*.

IV-C. Calidad de la Recomendación

Para analizar la calidad de la predicción de nuestro sistema lo hemos comparado con otros sistemas en dos aspectos. Por un lado, hemos analizado si es capaz de predecir la valoración asociada a cada reseña y, por el otro, si es capaz de predecir también el restaurante al que pertenece. Comparamos nuestro sistema, XRec (eXplainable Recommender), con sistemas sencillos como son el predecir siempre la media de las valoraciones (sistema *media*) o decidir a qué restaurante pertenece una reseña en función de la popularidad de los restaurantes (sistema *popularidad*). También lo hemos comparado con sistemas que tratan de predecir la valoración a partir únicamente del texto codificado mediante BoW (sistema bow2val) o mediante una red LSTM (sistema *lstm2val*) y con otros sistemas que tratan de predecir el restaurante codificando el texto como BoW (sistema *bow2rest*) o utilizando una red LSTM (sistema *lstm2rest*). En todos los casos se realizaron múltiples pruebas sobre el conjunto de validación para ajustar los hiperparámetros de cada uno de estos sistemas.

En la Tabla I se muestran los resultados alcanzados por los diferentes sistemas a la hora de predecir la valoración correspondiente a cada reseña. Podemos apreciar en los resultados que nuestro sistema, XRec, obtiene un error absoluto medio del mismo orden que el obtenido por el sistema lstm2val,

Tabla I
ERROR CUADRÁTICO Y ERROR ABSOLUTO MEDIO OBTENIDO POR LOS DIFERENTES SISTEMAS

Sistema	ECM	EAM
media	1.441	0.915
bow2val	0.722	0.629
lstm2val	0.426	0.483
XRec	0.442	0.507

Tabla II
PRECISIÓN@{1,5,10} OBTENIDA POR LOS DIFERENTES SISTEMAS EN PORCENTAJE

Sistema	P@1	P@5	P@10
popularidad	6.77	17.63	25.68
bow2rest	30.72	55.24	66.07
lstm2rest	34.18	57.30	69.49
XRec	29.44	54.75	65.49

considerablemente menor que el obtenido por bow2val y muy distanciado del que se obtendría prediciendo siempre la valoración media observada en el conjunto de entrenamiento. No debemos olvidar que nuestro sistema es el único de la comparativa que a la vez que trata de predecir correctamente la valoración, también está tratando de predecir el restaurante al que pertenece la reseña.

En la Tabla II, podemos ver el comportamiento de los sistemas a la hora de predecir el restaurante. Si se predice siempre el restaurante más popular (con mayor número de reseñas), se acertaría el 6.77 % de las veces. Si se predicen los 5 restaurantes más populares, entonces se acierta el 17.63 % y si son 10 el acierto asciende al 25.68 %. Estos resultados están muy alejados de los obtenidos por nuestro sistema, XRec, y el sistema bow2rest que presentan aciertos en torno al 30 %, 55 % y 66 % prediciendo 1, 5 y 10 restaurantes respectivamente. El sistema que mejor se comporta a la hora de predecir el restaurante es lstm2rest que, centrándose únicamente en acertar el restaurante al que pertenece la reseña, obtiene unos resultados algo mejores que los obtenidos por nuestro sistema. Sin embargo, como veremos en la siguiente sección, lstm2rest no es válido para ofrecer explicaciones transparentes.

IV-D. Explicación de la Recomendación

Como comentamos en las Sección II, muchos de los sistemas de recomendación que tratan de explicar sus recomendaciones se limitan simplemente a ofrecen justificaciones, habitualmente basadas en cuestiones de cercanía entre usuarios, productos o reseñas [11]. Utilizar aprendizaje profundo, como es el caso de las redes LSTM, nos permite obtener codificaciones de los textos en un nuevo espacio sobre el cual podremos calcular distancias y obtener justificaciones de las recomendaciones efectuadas. Sin embargo, estas justificaciones no permiten a los usuarios entender el funcionamiento del sistema y, por tanto, ser capaces de escrutar la recomendación.

En cambio, nuestro sistema realiza el aprendizaje de la valoración y del restaurante de manera simultánea y, además, incluye en su arquitectura el aprendizaje de la matriz $\mathbf{W}$ (2), que recoge la importancia de los 300 términos seleccionados para cada uno de los restaurantes.

Basándonos en la matriz $\mathbf{W}$, nuestro sistema puede proporcionar explicaciones que acompañen a la recomendación. Un ejemplo de recomendación con explicación podemos verlo en la Figura 3. Partiendo de la consulta *"Quiero comer un arroz con bogavante y con buenas vistas"*, nuestro sistema puede mostrar los 3 restaurantes (o el número que se desee) que mayor probabilidad tienen de ser asociados con ese texto. Además, cuando se coloca el foco sobre unos de los restaurante recomendados, se pueden resaltar los términos de la consulta mostrando la importancia que han tenido para que ese restaurante en concreto fuese recomendado. La intensidad del resaltado dependerá del valor que tenga el término para el restaurante seleccionado. También vemos en la Figura 3 que nuestro sistema ha recomendado el restaurante *Los Nogales* porque 4 términos de la búsqueda son importantes para definir ese restaurante, siendo los más importantes *"vistas"* y *"bogavante"*. Asimismo, se muestra un mapa de palabras en el que se pueden ver los 10 términos de la matriz $\mathbf{W}$ con mayor importancia para ese restaurante. Si observamos ese mapa para el segundo restaurante vemos que entre los términos que aparecen en el mapa figuran *"bogavante"*, *"playa"* y *"Lorenzo"*, dando a entender que si nos decidimos por el restaurante *Bellavista* podríamos comer el arroz con bogavante con vistas a la playa de San Lorenzo de Gijón.

Estas explicaciones, que permiten al usuario comprender el funcionamiento del sistema y que, por tanto, lo hacen transparente, hacen posible también que el usuario sepa cómo reformular una consulta con el fin de obtener una recomendación que se ajuste a sus requerimientos. En nuestro ejemplo, viendo el resultado mostrado en la Figura 3, si quisiésemos vistas a la playa podríamos reformular la pregunta y elegir un restaurante que resalte el término *"playa"*.

Además, en la parte superior derecha de la figura se muestra la valoración que el sistema estima que el usuario otorgaría al restaurante en caso de acudir a alguno de los recomendados, que en este ejemplo sería 4.5 estrellas.

V. CONCLUSIONES Y TRABAJO FUTURO

En este artículo hemos mostrado una manera de ofrecer explicaciones basadas en los textos de las reseñas que se comparten en Internet. A partir de esas reseñas hemos ideado un sistema de recomendación transparente que ofrece explicaciones basadas en los parámetros aprendidos durante el entrenamiento y que sirven para definir los aspectos importantes de un producto. Las explicaciones proporcionadas posibilitan a los usuarios comprender cómo el sistema elabora la recomendación y, además, permiten que el sistema sepa cómo reformular su consulta para obtener una mejor recomendación.

Nuestro sistema aprende de manera simultánea la valoración asociada a la reseña y el producto al que pertenece obteniendo un rendimiento similar al que se consigue realizando los dos aprendizajes de manera independiente.

Como en la arquitectura de nuestro sistema no intervienen directamente los usuarios, nuestro sistema podría resultar



Figura 3. Ejemplo de recomendación. En la parte superior aparece el texto de la consulta hecha por un usuario. Al situar el ratón sobre cada uno de los restaurante recomendados se resaltan (con diferentes intensidades) aquellas palabras de la consulta que más influencia han tenido para recomendar el restaurante seleccionado. Además, la imagen de cada restaurante se acompaña con un mapa de palabras que muestra la relevancia de cada término para cada restaurante (previamente aprendida por nuestro sistema). En la parte superior también se muestra la valoración que el sistema cree que le otorgaría el usuario a la experiencia que viviría si va a alguno de los 3 restaurantes recomendados (en este caso 4.5 estrellas).

también muy útil para ofrecer recomendaciones en situaciones de *arranque en frío*, cuando no haya un histórico del usuario.

Como trabajo futuro, planteamos un tratamiento más exhaustivo de los términos mediante la eliminación del género y número de las palabras o mediante el análisis de la similitud de las codificaciones obtenidas mediante el algoritmo word2vec. El uso del word2vec en este sentido nos permitiría realizar agrupaciones de términos con el mismo significado, como por ejemplo los términos *bugre* y *bogavante* que aparecen en el mapa de palabras del restaurante *Los Nogales* y que significan lo mismo puesto que *bugre* es el nombre que recibe el *bogavante* en Asturias. Además, introduciremos el uso de bigramas para tratar el tema de la negación, ya que hemos detectado que el sistema confunde a veces sentimientos positivos y negativos sobre determinados términos.

Agradecimientos

Agradecemos a NVIDIA Corporation la donación de la GPU Titan Xp utilizada en esta investigación.

Referencias

[1] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, "Grouplens: An open architecture for collaborative filtering of netnews," in *Proceedings of the 1994 ACM conference on Computer supported cooperative work*, pp. 175–186, 1994.

[2] F. Ricci, L. Rokach, and B. Shapira, "Introduction to recommender systems handbook," in *Recommender systems handbook*, pp. 1–35, Springer, 2011.

[3] B. Pathak, R. Garfinkel, R. D. Gopal, R. Venkatesan, and F. Yin, "Empirical analysis of the impact of recommender systems on sales," *Journal of Management Information Systems*, vol. 27, no. 2, pp. 159–188, 2010.

[4] S. Banker and S. Khetani, "Algorithm overdependence: How the use of algorithmic recommendation systems can increase risks to consumer well-being," *Journal of Public Policy & Marketing*, vol. 38, no. 4, pp. 500–515, 2019.

[5] D. Monroe, "Ai, explain yourself," *Communications of the ACM*, vol. 61, no. 11, pp. 11–13, 2018.

[6] Y. Zhang and X. Chen, "Explainable recommendation: A survey and new perspectives," *arXiv preprint arXiv:1804.11192*, 2018.

[7] N. Tintarev and J. Masthoff, "A survey of explanations in recommender systems," in *2007 IEEE 23rd international conference on data engineering workshop*, pp. 801–810, IEEE, 2007.

[8] A. I. Schein, A. Popescul, L. H. Ungar, and D. M. Pennock, "Methods and metrics for cold-start recommendations," in *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 253–260, 2002.

[9] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[10] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[11] N. Tintarev and J. Masthoff, "Evaluating the effectiveness of explanations for recommender systems," *User Modeling and User-Adapted Interaction*, vol. 22, no. 4-5, pp. 399–439, 2012.

[12] P. Pu, L. Chen, and R. Hu, "Evaluating recommender systems from the user's perspective: survey of the state of the art," *User Modeling and User-Adapted Interaction*, vol. 22, no. 4-5, pp. 317–355, 2012.

[13] A. Almahairi, K. Kastner, K. Cho, and A. Courville, "Learning distributed representations from reviews for collaborative filtering," in *Proceedings of the 9th ACM Conference on Recommender Systems*, pp. 147–154, 2015.

[14] F. Costa, S. Ouyang, P. Dolog, and A. Lawlor, "Automatic generation of natural language explanations," in *Proceedings of the 23rd international conference on intelligent user interfaces companion*, pp. 1–2, 2018.

[15] A. Bartoli, A. De Lorenzo, E. Medvet, D. Morello, and F. Tarlao, "Best dinner ever!!!: Automatic generation of restaurant reviews with lstm-rnn," in *2016 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, pp. 721–724, IEEE, 2016.

[16] R. Zhao and K. Mao, "Fuzzy bag-of-words model for document representation," *IEEE transactions on fuzzy systems*, vol. 26, no. 2, pp. 794–804, 2017.

[17] Y. Shao, S. Taylor, N. Marshall, C. Morioka, and Q. Zeng-Treitler, "Clinical text classification with word embedding features vs. bag-of-words features," in *2018 IEEE International Conference on Big Data (Big Data)*, pp. 2874–2878, IEEE, 2018.

[18] H. Caselles-Dupré, F. Lesaint, and J. Royo-Letelier, "Word2vec applied to recommendation: Hyperparameters matter," in *Proceedings of the 12th ACM Conference on Recommender Systems*, pp. 352–356, 2018.

[19] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

Método XAI basado en agrupamiento para la explicación de errores en clasificación

Aurora Ramírez
Dpto. de Informática y Análisis Numérico
Universidad de Córdoba, Córdoba 14071
Email: aramirez@uco.es

Sebastián Ventura y José Raúl Romero
Dpto. de Informática y Análisis Numérico,
Instituto Interuniversitario de Investigación en
Data Science and Computational Intelligence (DaCSI)
Universidad de Córdoba, Córdoba 14071
Email: {sventura,jrromero}@uco.es

***Resumen*—Los modelos predictivos obtenidos por medio de aprendizaje automático están alcanzando altas cotas de precisión, pero no son completamente infalibles. Los errores de clasificación pueden tener gran repercusión en ciertas aplicaciones, generando desconfianza si no es posible entender por qué suceden. La inteligencia artificial explicable (XAI), cuyos métodos están dirigidos a explicar el comportamiento de los modelos predictivos y sus decisiones, puede ayudar a comprender las características de las instancias mal clasificadas. Este trabajo propone un método XAI basado en un algoritmo de agrupamiento para identificar los diferentes tipos de errores que comete cualquier algoritmo de clasificación binaria. A continuación, se utilizan métodos XAI locales sobre las instancias más representativas para analizar qué características son las más influyentes. Los experimentos con diferentes clasificadores y conjuntos de datos ilustran cómo el método propuesto es capaz de encontrar diferentes grupos de instancias que inducen a errores, permitiendo al usuario detectar situaciones en las que las predicciones deberían ser revisadas.**

***Index Terms*—Inteligencia artificial explicable, clasificación binaria, errores, agrupamiento**

I. Introducción

El auge de los sistemas basados en inteligencia artificial ha propiciado la necesidad de analizar cómo funcionan sus métodos y qué implicaciones tienen sus resultados. Muchos modelos de decisión construidos automáticamente son opacos o difíciles de entender, lo cual genera desconfianza en sus predicciones. En este contexto surge XAI (*eXplainable Artificial Intelligence*), que promulga la construcción de modelos capaces de explicar su proceso de decisión y las razones que subyacen a sus predicciones con el objetivo de favorecer la transparencia, aumentar la credibilidad y reducir sesgos [1].

Una de las causas que genera falta de confianza es el hecho de que las predicciones obtenidas por medio de aprendizaje automático (ML, *Machine Learning*) no son infalibles [2]. En contextos como la medicina, una predicción errónea puede tener graves consecuencias, por lo que es importante entender cuándo y por qué un modelo predictivo comete errores. XAI puede aportar una visión novedosa, identificando si dichos errores se deben a características concretas de los datos o son propiciados por el propio proceso de aprendizaje.

Los métodos XAI pueden orientarse a explicar el *modelo* predictivo o sus *salidas* [3]. En el primer caso, lo habitual es que el método XAI tenga alcance *global*, frente a la explicación de salidas con carácter *local*. A su vez, el método XAI puede ser *específico* o *agnóstico*, en función de si depende o no del tipo de modelo predictivo. Finalmente, los métodos XAI pueden basarse en inducir, deducir o contrastar información obtenida de los modelos predictivos y/o sus predicciones. Un mecanismo inspirado en cómo los humanos razonan consiste en la generación de ejemplos y contraejemplos [4], pues permiten acotar la explicación a casos "prototipo". Estudios con usuarios han demostrado que este tipo de explicaciones contribuyen a comprender el funcionamiento de ML [2].

A pesar de su relevancia y aplicabilidad, existen pocos métodos XAI orientados a explicar errores y los que hay son específicos del dominio de aplicación [5], focalizados en regresión [2] o que solo ofrecen información local [6]. En este artículo proponemos un método XAI agnóstico que combina técnicas de agrupación, para analizar los resultados del clasificador de forma global, con técnicas de explicación de instancias, para analizar de forma local instancias representativas entre aquellas mal clasificadas.

Las principales contribuciones de nuestro trabajo son: 1) el uso de un algoritmo de agrupamiento para la identificación de instancias prototipo, 2) una comparación entre métodos XAI locales para estudiar la influencia de las características en los errores prototipo identificados, y 3) un estudio experimental que analiza los errores de clasificación cometidos por diferentes algoritmos sobre varios conjuntos de datos. Nuestros resultados demuestran que el algoritmo de agrupamiento es eficiente y se ajusta al tipo de análisis requerido (cuando pocas instancias son erróneas). Además, los prototipos generados responden de manera diferente al análisis local, reflejando que no todos los errores cometidos por un algoritmo de clasificación tienen por qué responder a las mismas causas.

El resto del artículo se estructura como sigue. La sección II presenta trabajos relacionados. La sección III describe el método XAI propuesto. El marco experimental se detalla en la sección IV, seguido de los resultados y su discusión en la sección V. La sección VI recoge las conclusiones.

II. Trabajo relacionado

Nuestra propuesta está relacionada con trabajos que analizan las salidas de modelos de clasificación y, en especial, los

Financiado por UCO y Feder (UCO-FEDER 18 REF.1263116 MOD), Min. Ciencia e Innovación (PID2020-115832GB-I00), J. Andalucía (DOC_00944)

errores cometidos. Un primer trabajo aplica razonamiento simbólico sobre las salidas de un clasificador de imágenes con el objetivo de analizar las regiones mal identificadas [5]. Los objetos inferidos en la imagen representan la explicación, pero sin asociar aquellos mal identificados con el funcionamiento del clasificador. En el contexto de una aplicación real de ventas, se ha propuesto un método XAI local para explicar las instancias que presentan errores desmesurados durante la fase de prueba [2]. En este caso, se establece un umbral para filtrar las instancias, tras lo cual se estudia la correlación entre la predicción y cambios en las características.

Otros dos trabajos comparten ciertas características con nuestra propuesta. En primer lugar, Rizzi et al. aplican métodos XAI locales sobre cada una de las instancias mal clasificadas para así determinar las características que más influyen en las predicciones erróneas [6]. A continuación, se estudia la frecuencia con la que cada característica aparece en las instancias mal clasificadas. Tal y como explican los autores, combinar explicaciones locales para obtener explicaciones globales todavía requiere de un mayor estudio, aparte de que aproximar el comportamiento global a partir de métodos locales como LIME puede inducir a interpretaciones erróneas [7]. Por otra parte, Kim et al. han abordado las explicaciones basadas en ejemplos y contraejemplos [4]. Los autores plantean la necesidad de identificar partes del conjunto de datos que difieren de los prototipos seleccionados, utilizando para ello un algoritmo voraz. Su propuesta se basa en analizar la similitud entre instancias para encontrar las que difieren en mayor medida, aunque para ello utilizan todo el conjunto de datos. En sus experimentos comparan el algoritmo de extracción de prototipos con un algoritmo de agrupamiento (k-medoids).

III. Descripción del método

La Figura 1 muestra un esquema general del método XAI propuesto, el cual consta de tres fases: 1) entrenar y validar un clasificador, extrayendo las instancias que el clasificador predice erróneamente, 2) agrupar las instancias sobre las que el clasificador falla en su predicción, y 3) explicar las instancias prototipo sobre las que se han cometido los errores de clasificación. A continuación, se explica cada fase en detalle.

III-A. Entrenamiento y prueba

Esta fase engloba el procedimiento habitual para entrenar y probar un modelo de clasificación. Tras aplicar el preprocesado necesario (no representado en la figura), deben separarse los datos para entrenamiento (aplicando validación cruzada u otra estrategia) y prueba. El clasificador puede ser entrenado por medio de cualquier técnica (redes neuronales, máquinas de vector soporte, etc.), ya que el método propuesto es agnóstico.

Una vez construido el clasificador, se comprueba si el clasificador acierta o falla en su predicción sobre la partición de prueba. En caso de fallo, la instancia se identifica como falso positivo (FP) o como falso negativo (FN). Aunque aquí se ilustra el caso de un problema de clasificación binaria, el método puede ser fácilmente adaptado a problemas multi-clase separando los errores por clases.

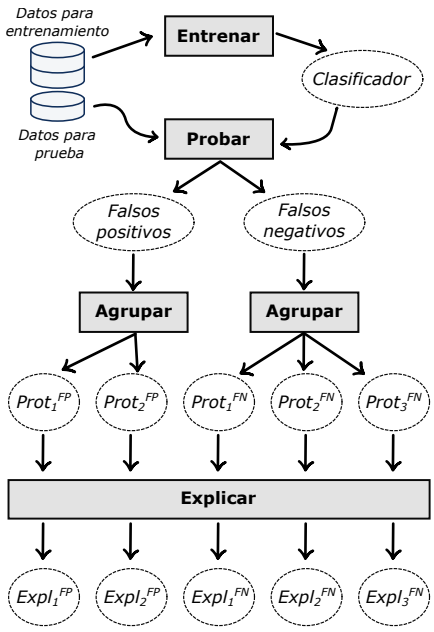


Figura 1: Visión general del método XAI.

III-B. Agrupación de instancias

En esta fase comienza el proceso de análisis de errores, el cual es equivalente entre FP y FN. El objetivo es encontrar grupos de instancias mal clasificadas para detectar las zonas de la distribución de los datos en las cuales el clasificador tiende a fallar. Un segundo objetivo es reducir la cantidad de instancias que se presentarán al usuario durante la fase de explicación, de forma que el análisis se centre en las más representativas. Además, los prototipos deben corresponderse con instancias del conjunto de datos para que le resulten realistas al usuario.

En base a los dos objetivos expuestos, en esta fase se emplea un algoritmo de agrupamiento llamado *Affinity Propagation* (AP) [8]. Este algoritmo posee dos características que lo hacen apropiado para los objetivos planteados: 1) no necesita preconfigurar el número de grupos a descubrir y 2) localiza, para cada grupo, la instancia más representativa a la que denomina *exemplar*. AP se basa en una estrategia de ”paso de mensajes” entre las instancias para elegir cuál de ellas es el mejor *exemplar* para otras instancias, y determinar a qué *exemplar* debe asociarse cada una de las instancias restantes. El proceso se repite iterativamente hasta lograr grupos de instancias estables. Las reglas de actualización de los *exemplars* se basan en las siguientes medidas:

1. Similitud ($s(i,j)$), representada como una función de distancia entre dos instancias i y j. Por defecto, se considera la distancia euclídea.
2. Responsabilidad ($r(i,j)$), que indica el grado en que la instancia j es un buen *exemplar* para la instancia i considerando el resto de posibles *exemplars* para i.
3. Disponibilidad ($a(j)$), que calcula, para un *exemplar* j, el grado en que es representativo del resto de instancias.

AP tiene complejidad cuadrática respecto al número de instancias, por lo que puede ser costoso computacionalmente. No obstante, el subconjunto de instancias mal clasificadas será pequeño en comparación al total de instancias, por lo que el algoritmo es adecuado para nuestro propósito. Por otro lado, AP es sensible a varios parámetros, pudiendo no llegar a converger. Sus tres parámetros son: $damping$, un valor entre 0 y 1 que regula el paso de mensajes; max_iter, que fija el número máximo de iteraciones; y $convergence\_iter$, que determina el máximo número de iteraciones antes de reajustar el número de grupos. Ante la posible falta de convergencia, nuestra experimentación incluye un análisis de la influencia de estos parámetros (Sección V-B). No obstante, la no convergencia de AP puede interpretarse como la existencia de un único grupo formado por todas las instancias mal clasificadas, eligiendo uno o varios *exemplars* de manera aleatoria [4].

Como resultado de esta fase se obtiene una separación de instancias FP y FN en k grupos (el valor de k no tiene por qué ser el mismo en ambos casos). Para cada grupo, se dispone además de una instancia *exemplar* que representa el prototipo sobre el cual se van a generar las explicaciones locales.

III-C. Explicación de errores

La última fase consiste en la generación de una explicación para cada uno de los prototipos de error identificados en el paso anterior. Puesto que cada prototipo es una instancia real del conjunto de datos, su explicación puede generarse por medio de un método XAI local para explicación de salidas. En concreto, se va a estudiar el funcionamiento de tres métodos:

- *LIME* [9]: este método construye un modelo lineal simple que se aproxima al modelo predictivo completo en el entorno de la instancia. Para ello genera instancias en su vecindad mediante un método basado en perturbación.
- *SHAP* [10]: es un método basado en la teoría de juegos que busca encontrar una distribución óptima de la importancia que tiene cada característica en la predicción devuelta por el modelo. Se trata de un método fundamentado en una base teórica que unifica otras propuestas.
- *BreakDown* [11]: este método también calcula la influencia de cada característica en la predicción obtenida. Su principal diferencia es que sumando la atribución que recibe cada característica se obtiene el valor de la predicción, lo cual facilita su interpretación.

IV. Marco experimental

Esta sección describe los experimentos que estudiarán el rendimiento del método propuesto. A continuación, se detallan los conjuntos de datos utilizados y la parametrización de los algoritmos.

El método XAI y la experimentación se han implementado en Python utilizando las siguientes librerías: *scikit-learn* [12], para el preprocesado de datos y la ejecución de los algoritmos de clasificación y agrupamiento; y *dalex* [13] para la ejecución de métodos XAI locales, el cual a su vez requiere el paquete *lime*[1] para la ejecución de dicho método. La experimentación se ha ejecutado en una máquina con Debian 8, ocho núcleos Intel Core i7-2600, CPU a 3.4 GHz y 16 GB de RAM.

[1]https://github.com/marcotcr/lime (accedido: 21/05/2021)

Tabla I: Conjuntos de datos y sus características

Nombre	Num. Características	Num. Instancias
Breast cancer	9	277
Column	6	310
Credit	15	653
Diabetes	8	768

Tabla II: Algoritmos y sus parámetros

Parámetro	Valores
Random Forest (RF)	
n_estimators	50, 100, 500, 1000
max_depth	10, 25, 50, ilimitado
max_features	$\sqrt{NC}$ (*auto*), todas
min_samples_split	2, 5, 10
min_samples_leaf	1, 2, 5
Support Vector Machine (SVM)	
kernel	lineal, radial, sigmoide
gamma	$1/(NC * var(d))$ (*scale*), $1/NC$ (*auto*)
max_iter	500, 1000, 5000, 10000, ilimitado
Affinity Propagation (AP)	
damping	0.5, 0.7, 0.9
max_iter	100, 200, 500
convergence_iter	5, 15, 30

IV-A. Diseño experimental

Como paso previo a la experimentación del método XAI, es necesario obtener instancias mal clasificadas para su análisis. Por ello, primero estudiaremos las salidas de dos algoritmos de clasificación bajo dos configuraciones de parámetros (por defecto y optimizados), con el objetivo de comprobar si afectan al número de errores cometidos. A continuación, se plantean dos experimentos para estudiar la influencia de los elementos que intervienen en cada fase del método XAI:

1. *Experimento #1*. El objetivo es estudiar el rendimiento del algoritmo de agrupamiento, con especial énfasis en la configuración de sus parámetros para garantizar la convergencia. Se considerarán los resultados sobre la salida del clasificador con parámetros por defecto y optimizados, a fin de establecer diferencias en cuanto al número de grupos identificados y sus características.
2. *Experimento #2*. El objetivo es comparar los métodos descritos en la sección III-C. Dada la limitación de espacio, se utiliza un único clasificador sobre un conjunto de datos según lo analizado en los experimentos previos.

IV-B. Conjuntos de datos

La Tabla I detalla las características de los conjuntos de datos seleccionados. Todos ellos tratan un problema de clasificación binaria, aunque varían en cuanto al dominio de aplicación, número de características (#C) y número de instancias (#I). Algunos de estos conjuntos de datos son habituales en experimentos con métodos XAI [14], [15].

Los conjuntos de datos han sido divididos en partición de entrenamiento (75 %) y de prueba (25 %). La partición de entrenamiento es la utilizada para el estudio de parámetros, aplicando a su vez validación cruzada estratificada con 10

Tabla III: Resultados de los clasificadores sobre la partición de validación

Conjunto de datos	RF_d				RF_p				SVM_d				SVM_p			
	Acc	F1	FP	FN	Acc	F1	FP	FN	Acc	F1	FP	FN	Acc	F1	FP	FN
Breast cancer	0.7571	0.5143	6	11	0.7500	0.4138	3	14	0.7000	0.1600	3	18	0.6857	0.4762	12	10
Column	0.7692	0.8421	13	5	0.7692	0.8393	12	6	0.8077	0.8649	13	5	0.7564	0.8224	12	6
Credit	0.8902	0.8784	9	9	0.8902	0.8816	11	7	0.8293	0.8250	20	8	0.8780	0.8734	15	5
Diabetes	0.7969	0.6829	14	25	0.7969	0.6829	14	25	0.7760	0.6504	16	27	0.7708	0.6393	16	28

particiones. La partición de prueba se utiliza para evaluar el clasificador resultante del estudio de parámetros (o con su configuración por defecto), y es de donde se extraen las instancias mal clasificadas para su posterior análisis.

IV-C. Algoritmos y métricas de rendimiento

Para la construcción de clasificadores se utilizan *random forest* (RF) y *support vector machine* (SVM). La Tabla II muestra los parámetros que se van a estudiar para cada algoritmo. El valor subrayado corresponde a la configuración por defecto del algoritmo en *scikit-learn*. Se han establecido valores tanto superiores como inferiores a los valores por defecto, salvo que estos correspondan al mínimo o máximo teórico. Para el caso de SVM, el parámetro *max_iter* no se utiliza en la configuración por defecto, pero se ha incluido en el estudio de parámetros para comprobar si se puede reducir el tiempo de ejecución sin que afecte a la precisión del algoritmo.

El proceso de optimización de parámetros consiste en combinar todos los valores de cada parámetro entre sí, procedimiento conocido como *grid search*. Para cada combinación de parámetros, se aplica la validación cruzada y se ejecutan 30 semillas aleatorias diferentes. De cada ejecución se extrae la matriz de confusión y las siguientes métricas de rendimiento: *accuracy*, *recall*, *precision* y *F1*. Aquella combinación con la media más alta en la métrica F1 se elige como la mejor.

Para la segunda fase, se van a comparar 27 combinaciones de los tres parámetros del algoritmo AP (ver Tabla II). Para cada una de ellas se realizan 30 ejecuciones con distintas semillas aleatorias. Los resultados se analizan en base a las siguientes métricas: número de grupos, *Silhouette Coefficient* (SC), *Calinski-Harabasz Index* (CHI) y *Davies-Bouldin Index* (DBI). Se han elegido estas tres métricas porque no requieren la existencia de clases etiquetadas con las que comparar la agrupación realizada. Es decir, se basan en analizar la dispersión de los grupos creados. SC y CHI son métricas a maximizar, mientras que DBI debe ser minimizada.

V. Resultados y discusión

Esta sección presenta los resultados de la experimentación, discutiendo a su vez los aspectos más relevantes del método.

V-A. Análisis de errores en clasificación

Como punto de partida del método XAI propuesto, es conveniente analizar cuántos errores cometen los clasificadores bajo diferentes configuraciones de parámetros, y de qué tipo (FP o FN) son dichos errores. Esta comprobación es apropiada en XAI, pues es habitual que su audiencia objetivo sean usuarios no expertos en ML que tienden a utilizar las configuraciones por defecto de los algoritmos.

La Tabla III recoge las métricas de rendimiento para cada algoritmo sobre la partición de prueba, tanto con la configuración por defecto (RF_d y SVM_d) como para la mejor combinación de parámetros (RF_p y SVM_p). Centrándonos en RF, ambas configuraciones coinciden en el número total de instancias mal clasificadas (FP+FN). Para dos conjuntos de datos (*Breast cancer* y *Column*), la configuración optimizada reduce el número de FP, pero a costa de aumentar el de FN en igual proporción. Para el caso de SVM, limitar el número de iteraciones conlleva una reducción en la fiabilidad del clasificador para los conjuntos *Column* y *Diabetes*. No obstante, para *Credit* se han reducido tanto los FP como los FN, mientras que para *Breast cancer* se mejora $F1$ gracias a un menor número de FN.

En general, se puede afirmar que la optimización de parámetros puede resultar ventajosa en ciertos conjuntos de datos, pero que la configuración por defecto tiene un rendimiento similar. En el contexto de nuestra experimentación, no realizar una optimización de parámetros no va a repercutir significativamente en el número de errores a analizar en las siguientes fases del método para los conjuntos de datos seleccionados.

V-B. Análisis del algoritmo de agrupamiento

Una vez identificado el subconjunto de FP y FN para cada algoritmo y conjunto de datos, podemos estudiar la influencia de los parámetros del algoritmo AP. Las tablas IV y V recogen el número de grupos (#G) y las medidas de evaluación de los grupos (SC, CHI y DBI) para las instancias mal clasificadas por RF y SVM, respectivamente. Los valores representan la media de las 27 configuraciones de AP (ver Tabla II), junto a la desviación estándar. Para cada algoritmo, se diferencia entre FP y FN, así como según su configuración de parámetros.

Un primer hecho a destacar es que AP no proporciona una división en grupos ($\#G = 1$) para las instancias FN de los tres conjuntos de datos con menos instancias. El número de FN varía entre 5 y 18 instancias, mientras que para *Diabetes* es igual o superior a 25. No obstante, AP sí propone una división en grupos para todos los casos de FP, aún cuando el número de FP es inferior al de FN. Esto parece indicar que existe mayor diversidad entre las instancias negativas, esto es, podrían haber sido clasificadas como positivas por diferentes motivos. Por el contrario, todas las instancias positivas clasificadas como negativas podrían responder a un mismo patrón. Además, para el caso de los FP, el número de grupos sí aumenta en función del número de instancias como cabría esperar.

Por otro lado, se aprecia una diferencia respecto al agrupamiento de FN para el conjunto de datos *Diabetes*. Para RF, los 25 FN se agrupan en 4-6 grupos, mientras que los 27-28 FN

Tabla IV: Resultados del algoritmo de agrupamiento para errores de RF

	Conjunto de datos	RF_d				RF_p			
		#G	SC (↑)	CHI (↑)	DBI (↓)	#G	SC (↑)	CHI (↑)	DBI (↓)
FP	B. Cancer	2.08±0.03	0.15±0.00	1.85±0.00	0.42±0.00	2.18±0.04	0.16±0.00	1.97±0.00	0.41±0.00
	Column	4.00±1.41	0.19±0.03	6.40±0.62	0.85±0.29	3.44±0.83	0.25±0.05	5.21±0.58	0.75±0.18
	Credit	3.67±0.47	0.18±0.13	3.21±1.06	0.81±0.56	3.78±0.42	0.25±0.11	3.76±0.99	0.72±0.58
	Diabetes	3.89±1.10	0.26±0.01	4.44±0.72	0.68±0.01	4.56±1.26	0.30±0.00	7.27±0.00	0.69±0.00
FN	B. Cancer	1.00±0.00	-	-	-	1.00±0.00	-	-	-
	Column	1.00±0.00	-	-	-	1.00±0.00	-	-	-
	Credit	1.00±0.00	-	-	-	1.00±0.00	-	-	-
	Diabetes	6.67±2.05	0.14±0.01	5.23±0.27	0.92±0.07	6.33±1.89	0.13±0.00	4.96±0.00	0.98±0.00

Tabla V: Resultados del algoritmo de agrupamiento para errores de SVM

	Conjunto de datos	SVM_d				SVM_p			
		#G	SC (↑)	CHI (↑)	DBI (↓)	#G	SC (↑)	CHI (↑)	DBI (↓)
FP	B. Cancer	2.14±0.05	0.05±0.00	1.23±0.00	0.52±0.00	3.22±0.63	0.18±0.01	3.20±0.18	1.08±0.10
	Column	3.78±0.42	0.29±0.12	8.18±2.59	0.52±0.41	3.78±0.42	0.29±0.12	8.18±2.59	0.52±0.41
	Credit	3.78±0.42	0.25±0.11	3.76±0.99	0.72±0.58	3.78±0.42	0.25±0.11	3.76±0.99	0.72±0.58
	Diabetes	4.56±1.26	0.30±0.00	8.30±0.00	0.70±0.00	4.56±1.26	0.30±0.00	8.30±0.00	0.70±0.00
FN	B. Cancer	1.00±0.00	-	-	-	1.00±0.00	-	-	-
	Column	1.00±0.00	-	-	-	1.00±0.00	-	-	-
	Credit	1.00±0.00	-	-	-	1.00±0.00	-	-	-
	Diabetes	4.73±2.73	0.18±0.00	5.92±0.00	1.16±0.03	4.73±2.73	0.18±0.00	5.92±0.00	1.16±0.03

Tabla VI: Influencia de las características en los prototipos (FP del algoritmo RF_p para el conjunto de datos *Diabetes*)

Característica	**LIME**					**SHAP**					**BreakDown**				
	P_1	P_2	P_3	P_4	P_5	P_1	P_2	P_3	P_4	P_5	P_1	P_2	P_3	P_4	P_5
preg	−	−	+	−	−	+	−	+	−	−	≈ 0	−	−	−	−
plas	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
pres	−	+	−	+	−	+	−	+	+	−	+	+	−	+	−
skin	−	−	+	−	+	−	−	+	−	+	+	+	+	−	+
insu	−	−	+	−	+	−	−	+	−	−	+	+	+	+	+
mass	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
pedi	−	−	+	+	+	−	−	+	+	+	−	−	+	+	+
age	+	+	+	+	−	+	+	+	+	−	+	+	+	+	−

cometidos por SVM se han dividido en 2-4 grupos. Aunque el número de FN es parecido en ambos casos, cada clasificador se equivoca en instancias con características diferentes y que, por tanto, se agrupan de distinta manera. Los valores de desviación estándar indican una mayor variabilidad entre las distintas ejecuciones de AP para este conjunto de datos.

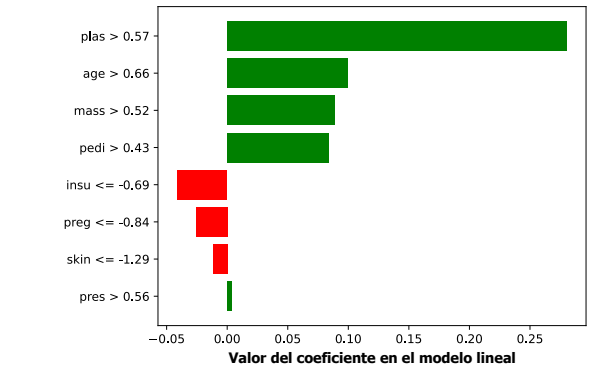
De entre las medidas de densidad, CHI es la que presenta una mayor variación entre conjuntos de datos con similar #G. En concreto, los grupos están menos cohesionados para el conjunto de datos *Credit* en comparación con *Column* y *Diabetes*, a pesar de que no existe una gran diferencia en #G. En general, los valores de desviación estándar indican que los resultados son muy estables entre las diferentes configuraciones de AP, incluso aunque #G varíe ligeramente.

Respecto a los parámetros de AP, cabe señalar que algunas configuraciones no han logrado converger para los conjuntos de datos *Breast cancer* y *Diabetes*. Esto sucede especialmente cuando $damping = 0.9$ y $convergence\_iter = 5$. En el resto de los casos, AP es bastante estable y su configuración por defecto es representativa del comportamiento general. El tiempo de ejecución ha sido siempre inferior a 0.25 segundos.

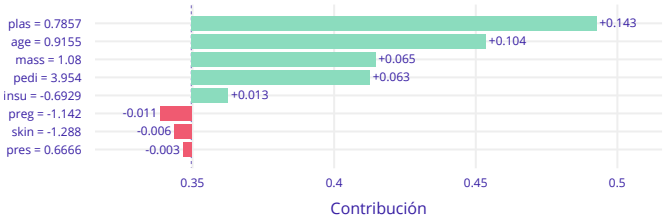
V-C. Análisis de las explicaciones locales

Para analizar las explicaciones asociadas a los prototipos, nos centramos en el algoritmo RF, por tener un comportamiento similar entre sus configuraciones, y el conjunto de datos *Diabetes*, para el que se produce un número relativamente alto de errores. Para la agrupación de errores, se ha ejecutado AP con parámetros por defecto. Como resultado, se obtienen cinco prototipos de FP y siete prototipos de FN. Por cuestiones de espacio, solo se realiza un análisis detallado de los FP. La probabilidad de pertenencia a la clase positiva devuelta por RF para cada FP es: $Prot_1^{FP} = 0.65$, $Prot_2^{FP} = 0.70$, $Prot_3^{FP} = 0.64$, $Prot_4^{FP} = 0.72$ y $Prot_5^{FP} = 0.56$. Estos valores ya nos indican que existen diferencias en cuanto al grado de error que comete RF en cada caso, pues no todos los valores están próximos a la barrera de decisión (0.5).

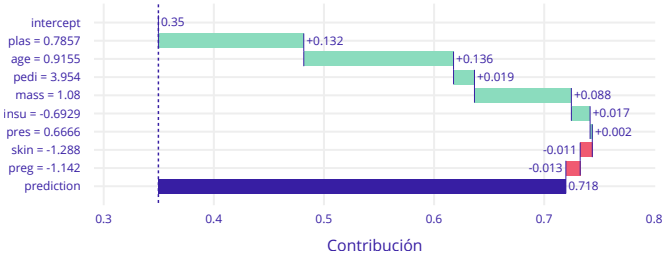
La Tabla VI muestra la influencia de las características en la predicción de cada prototipo (P), según *LIME*, *SHAP* y *BreakDown*. Dado que cada método calcula esta influencia de manera diferente, los valores concretos son poco representativos para la comparación y por tanto solo se indica el signo ($+$ si la característica contribuye positivamente a la predicción y $-$ en caso contrario). Observamos que cada prototipo tiene asociada una combinación diferente de características positivas y negativas. Este hecho confirma que son instancias que responden de manera diferente a la predicción. Por tanto, podemos concluir que AP ha separado instancias que, de cara al usuario final, no van a proporcionar explicaciones repetidas. Por otro lado, los métodos no siempre coinciden en el tipo de



(a) LIME



(b) SHAP



(c) BreakDown

Figura 2: Explicación local del prototipo $Prot_4^{FP}$

atribución (+ o -) de una misma característica para un mismo prototipo. La forma en que cada método analiza el entorno de vecindad y los elementos del modelo de decisión podrían ser la causa.

A modo de ejemplo, la Figura 2 muestra la salida de cada método para $Prot_4^{FP}$. Es el caso en el que existe mayor coincidencia entre los métodos y, además, es el prototipo que obtuvo mayor probabilidad de pertenencia a la clase positiva. El orden en el que contribuye cada característica es bastante similar entre los tres métodos, siendo las características *plas* y *age* las más influyentes. En resumen, la inclusión de métodos locales es capaz de aportar más información al usuario acerca de las características que influyen en los errores cometidos. No obstante, es necesario estudiar sus propiedades en más detalle para entender sus divergencias y el impacto que puedan tener en la interpretación de los resultados por parte del usuario.

VI. Conclusiones

En este trabajo se ha presentado un método XAI dirigido a explicar errores de clasificación en aprendizaje automático. Tras identificar las instancias clasificadas erróneamente, el algoritmo de agrupamiento (AP) permite extraer un subconjunto representativo de errores, que son luego estudiados mediante métodos XAI locales. La experimentación realizada muestra la eficacia del algoritmo AP para revelar diferencias entre los errores cometidos, mientras que la elección del método local puede resultar más condicionante de cara a interpretar las causas asociadas a cada prototipo de error encontrado.

Como trabajo futuro nos planteamos extender el estudio experimental para poder extraer conclusiones más generalizables. También es necesario estudiar en más detalle el comportamiento del algoritmo AP respecto a los FN, así como las diferencias observadas entre los métodos XAI locales.

Referencias

[1] A. Barredo Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. Garcia, S. Gil-Lopez, D. Molina, R. Benjamins, R. Chatila, and F. Herrera, "Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI," *Information Fusion*, vol. 58, pp. 82–115, 2020.

[2] A. Lucic, H. Haned, and M. de Rijke, "Why Does My Model Fail? Contrastive Local Explanations for Retail Forecasting," in *Proc. Conference on Fairness, Accountability, and Transparency (FAT*)*, 2020, pp. 90—98.

[3] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, and D. Pedreschi, "A Survey of Methods for Explaining Black Box Models," *ACM Computing Surveys*, vol. 51, no. 5, pp. 1–42, 2018.

[4] B. Kim, R. Khanna, and O. Koyejo, "Examples Are Not Enough, Learn to Criticize! Criticism for Interpretability," in *Proc. 30th International Conference on Neural Information Processing Systems (NIPS)*, 2016, p. 2288–2296.

[5] M. Alirezaie, M. Längkvist, M. Sioutis, and A. Loutfi, "A Symbolic Approach for Explaining Errors in Image Classification Tasks," in *IJCAI Workshop on Learning and Reasoning*, 2018.

[6] W. Rizzi, C. Di Francescomarino, and F. M. Maggi, "Explainability in predictive process monitoring: When understanding helps improving," in *Proc. International Conference on Business Process Management (BPM)*, 2020, pp. 141–158.

[7] D. Garreau and U. von Luxburg, "Explaining the Explainer: A First Theoretical Analysis of LIME," in *Proc. 23rd International Conference on Artificial Intelligence and Statistics*, 2020, pp. 1287–1296.

[8] B. J. Frey and D. Dueck, "Clustering by Passing Messages Between Data Points," *Science*, vol. 315, no. 5814, pp. 972–976, 2007.

[9] M. T. Ribeiro, S. Singh, and C. Guestrin, ""Why Should I Trust You?": Explaining the Predictions of Any Classifier," in *Proc. 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, 2016, pp. 1135—1144.

[10] S. M. Lundberg and S.-I. Lee, "A Unified Approach to Interpreting Model Predictions," in *Proc. 31st International Conference on Neural Information Processing Systems (NIPS)*, 2017, pp. 4768—4777.

[11] M. Staniak and P. Biecek, "Explanations of Model Predictions with live and breakDown Packages," *The R Journal*, vol. 10, no. 2, pp. 395–409, 2018.

[12] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[13] H. Baniecki, W. Kretowicz, P. Piatyszek, J. Wisniewski, and P. Biecek, "dalex: Responsible Machine Learning with Interactive Explainability and Fairness in Python," *arXiv:2012.14406*, 2020.

[14] B. P. Evans, B. Xue, and M. Zhang, "What's inside the Black-Box? A Genetic Programming Method for Interpreting Complex Machine Learning Models," in *Proc. Genetic and Evolutionary Computation Conference (GECCO)*, 2019, p. 1012–1020.

[15] A. White and A. d'Avila Garcez, "Measurable Counterfactual Local Explanations for Any Classifier," in *Proc. 24th European Conference on Artificial Intelligence (ECAI)*, 2020, pp. 1–7.

Revisión y análisis de conceptos en Inteligencia Artificial Explicable. Una aproximación a la unificación de la terminología

Ángela Escobar, Pedro González, María José del Jesus
Instituto Andaluz Interuniversitario en Data Science and Computational Intelligence (DaSCI)
Universidad de Jáen, Jaén, España
aem00007@red.ujaen.es, pglez@ujaen.es, mjjesus@ujaen.es

***Resumen*—Una de las principales dificultades a la hora de desarrollar modelos de inteligencia artificial explicable, es que no existe una definición única, común y aceptada en la bibliografía específica sobre algunos conceptos y métricas que faciliten cuantificar la interpretabilidad de un modelo. En este trabajo se ha realizado una revisión de la terminología aplicable en inteligencia artificial explicable en la bibliografía especializada, que permite, en algunos casos, aportar definiciones integradoras, y proponer la unificación y agrupación de conceptos. Es un análisis inicial en un área que debe afrontar muchos retos, entre ellos el debate, consenso y propuesta sobre conceptos y métricas objetivas que permitan cuantificar la explicabilidad de un modelo.**

***Index Terms*—inteligencia artificial explicable, XAI, explicabilidad, interpretabilidad, objetivos, propiedades**

I. Introducción

Cada vez son más los algoritmos de Inteligencia Artificial (IA) que conviven con nosotros en nuestro día a día y que son capaces de decidir o hacernos más fácil la toma de decisiones habituales. Por indicar sólo dos ejemplos, se podría pensar en los algoritmos implicados para la recomendación de contenidos en plataformas como Netflix [5] o en los desarrollados para la ayuda en el diagnóstico o severidad de la enfermedad del Covid19 [12]. Actualmente la aplicación de algoritmos de IA es transversal a la mayoría de actividades del ser humano, por lo que -al menos en algunos ámbitos, como p.e. el mencionado de la medicina, entre otros- es necesario entender cómo determinan la salida, ya que cuando hablamos del caso de Netflix, una mala recomendación puede hacernos perder dos horas como máximo de nuestro tiempo mientras que una decisión médica puede llegar a tener un riesgo mayor y es necesario entender porque el algoritmo presenta esta decisión. De forma paralela, los modelos IA desarrollados tienen un nivel de complejidad cada vez más alto lo que dificulta proporcionar esta explicación. En este contexto surge la Inteligencia Artificial Explicable (e**X**plainable **A**rtificial **I**nteligence, XAI).

Para comenzar, es necesario saber qué estamos tratando cuando hablamos de XAI. ¿Qué es XAI? XAI es el acrónimo de *Explainable Artificial Intelligence* o lo que es lo mismo, XAI es la forma en la que nos referimos a la explicabilidad de los métodos utilizados en IA y de los resultados obtenidos por los mismos. XAI no sólo se centra en la explicación de los métodos, algoritmos o técnicas utilizadas sino que también hace referencia a la explicación de los resultados hacia el usuario final.

En la bibliografía especializada se han propuesto diferentes definiciones de XAI, es por ello, que antes de seguir con este estudio, es positivo trabajar con una definición que aglutine y unifique todas estas definiciones.

De igual forma se han estudiado a lo largo del tiempo dentro del ámbito estrictamente del aprendizaje automático, como por ejemplo indica W. James Murdoch et al. [13] y fuera de él, diferentes conceptos, con diferente terminología, relacionados con la explicabilidad, interpretabilidad, sencillez, etc. de los modelos. El análisis, debate y unificación de términos y conceptos puede contribuir al desarrollo de modelos en XAI.

Un modelo de XAI estará caracterizado por diferentes propiedades o características, que serán objetivos a alcanzar en el proceso de diseño de los mismos. Sobre ellas, de nuevo, existen diferentes visiones, propuestas y definiciones.

En este trabajo se realiza un primer análisis de conceptos en XAI que pretende ayudar a contribuir a la unificación de la terminología.

Para ello, el artículo se organiza de la siguiente forma: en la sección "Inteligencia Artificial Explicable" se realizará una revisión de las diferentes definiciones dadas para llegar a una definición del mismo, en las siguientes secciones, "Conceptos XAI" y "Propiedades y objetivos en un modelo XAI" realizaremos el análisis de toda la terminología encontrada en la bibliografía especializada, unificando términos y realizando una propuesta de definición para cada uno de ellos. Finalmente en la sección de "'Conclusiones" se revisarán los resultados obtenidos durante el estudio y se propondrán nuevas líneas de trabajo futuras.

II. Inteligencia Artificial Explicable

La primera referencia a XAI viene dada por van Lent et al. en [14], y a esta referencia le siguen las dadas por Barocas et al. [2], Gunning [9], Gunning [10], Doshi-Velez et al. [4] y G. Vilone et al. [18]. En estos artículos se introduce de forma intuitiva el concepto XAI destancando la importancia

a que un modelo de IA sea explicable, pero no se llega a dar una definición del mismo. No es hasta Barredo et al. [3] donde encontramos una definición clara de este concepto: "Dada una audiencia, un modelo XAI es aquel que produce detalles o razones para que su funcionamiento sea claro o fácil de entender". Revisando esta definición, se plantea la duda de ¿cómo reportamos esta explicabilidad al usuario final? Para ello nos apoyamos en Mosheni et al. [17] que no define el término pero que indica que la explicación ha de presentarse al usuario con respecto a su nivel de conocimientos.

Partiendo de la definición dada por [3] y de la segregación de usuarios y formatos dada por [17] se puede plantear la siguiente definición de XAI: "Entendemos por un algoritmo de inteligencia artificial explicable a aquel algoritmo que es capaz de producir detalles o razones que haga entender y esclarecer su funcionamiento, así como que consiga dar una explicación de porqué se deriva a una toma de decisión final, dependiendo del tipo de usuario al que llega esa explicación." Podemos representar esta definición de forma visual basada en el ejemplo dado por [9] en la figura 1.

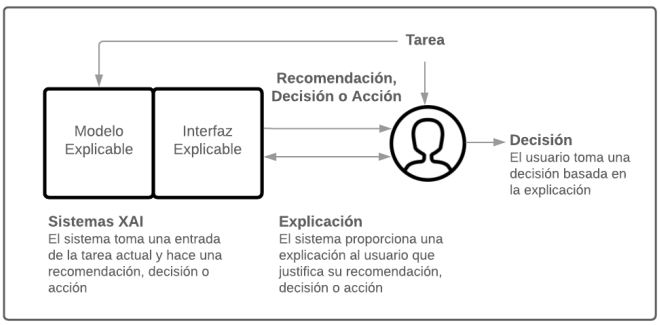


Figura 1. Definición visual de XAI basada en [9]

En la figura 1 podemos ver como un sistema XAI está formado por el modelo y por la interfaz explicable con la que el tomador de decisiones interactúa para tomar una decisión a partir de los resultados de este. Se empieza a ver como se da cabida a la forma en la que el modelo da sus resultados a partir de la interfaz explicable.

III. Conceptos XAI

En esta sección se revisan distintos conceptos relacionados con XAI, entre los que se encuentran inteligibilidad, comprensibilidad, transparencia, explicabilidad, interpretabilidad y explicabilidad post-hoc. En primer lugar se propone agrupar los conceptos en función de su aplicabilidad al modelo o al resultado obtenido por el mismo. A continuación se analizan los conceptos que se encuadran en cada una de las dos categorías, estudiándose las diferencias de definición entre los distintos autores y se propone una definición unificada de los mismos.

III-A. Conceptos aplicables al modelo

En este grupo analizaremos los conceptos relativos a la explicabilidad del algoritmo, de un modelo o su funcionalidad. Estos conceptos son: inteligibilidad, comprensibilidad y transparencia.

Inteligibilidad (understandability): Son varios los autores que hacen referencia a este concepto, entre ellos Barredo et al. [3], Montavon et al. [16] y Lipton [15]. En las referencias [3] y [16] se indica, de forma general, que un modelo es inteligible si es capaz de ser entendible su funcionalidad por un ser humano sin necesidad de profundizar en el mismo. Sin embargo, en [15] se equiparan los conceptos de interpretabilidad e inteligibilidad. Más adelante se abordará la definición de interpretabilidad, pero estos autores indican que para ellos un modelo inteligible ya es de por sí interpretable.

En función de las diferentes definiciones introducidas por estos autores, se puede entender que un modelo es inteligible si funcionalmente es entendible por un ser humano sin necesidad de profundizar en el funcionamiento interno del mismo.

Comprensibilidad (comprehensibility): Varios autores hablan del mismo, desde Freitas [7] que indica que este concepto es muy subjetivo ya que depende de los usuarios y del dominio de la aplicación que se tenga del mismo, pasando por Fernández et al. [8] que iguala este concepto al de interpretabilidad y terminando por Doshi-Velez et al. [6] y Barredo et al. [3] que indican que este concepto hace referencia a la capacidad de un algoritmo de representar el conocimiento aprendido de forma que sea comprensible para el ser humano.

En base a esto se podría establecer que un modelo es comprensible si tiene suficiente capacidad para representar de forma entendible para el ser humano el conocimiento aprendido sea cual sea la profundidad del mismo.

Transparencia (transparency): Encontramos dos autores que hacen referencia al término, por una parte Barredo et al. [3] que indica que un modelo es transparente si por si mismo es inteligible *(understandable)*. En el análisis se hace referencia a la descomposición por niveles del modelo presentado por Lipton [15]. En este último se plantean los siguientes tres niveles de descomposición para este concepto. Según Lipton [15] el modelo puede:

- ser totalmente transparente en todos sus procesos (simulabilidad, en inglés *simulatability*),
- sólo ser transparente de forma modular a partir de la descomposición del mismo (descomponibilidad, en inglés *decomposability*), o
- que sólo lo sea su algoritmo de entrenamiento (transparencia algorítmica, en inglés *algorithmic transparency*).

Visto todo lo anterior, se puede entender que un modelo es transparente cuando su modelo funcional es comprensible por sí mismo. Basándonos en los niveles de descomposición de esta propiedad indicados en [15], la cuestión a plantearnos ahora sería ¿consideramos que un modelo es totalmente transparente si cumple el nivel de simulabilidad o consideramos un modelo transparente sólo con cumplir uno de los tres niveles descritos? Para responder

a esta pregunta, nos centraremos en si estamos tratando con un modelo de caja blanca o caja negra. Si se trata de un modelo de caja negra, lo consideraremos transparente si sólo su algoritmo de entrenamiento es comprensible funcionalmente (transparencia algorítmica). En el caso de un modelo de caja blanca, para considerar que un modelo es transparente, debe de ser comprensible funcionalmente a nivel de modelo completo (simulabilidad), es decir, debe de ser comprensible funcionalmente de forma modular a partir de su descomposición a la vez que también lo es su algoritmo de entrenamiento.

Como se puede observar en las distintas definiciones analizadas hasta el momento, la diferencia que encontramos en cada uno de los conceptos es muy pequeña y su matiz diferenciador se encuentra en la profundidad modular o algorítmica en la que cada uno de estos conceptos es capaz de ser entendible para el ser humano. Es por ello, que nos planteamos la siguiente cuestión: ¿Se podría establecer una gradación de mayor a menor explicabilidad en estos conceptos XAI aplicables a nivel de modelo? Al encontrar este matiz diferenciador, es posible realizar esta gradación de conceptos que quedaría, de menos a más, de la siguiente forma:

- Inteligibilidad
- Comprensibilidad
- Transparencia
  - Transparencia Algorítmica
  - Descomponibilidad
  - Simulabilidad

La figura 2 muestra de forma visual la gradación de estos conceptos:

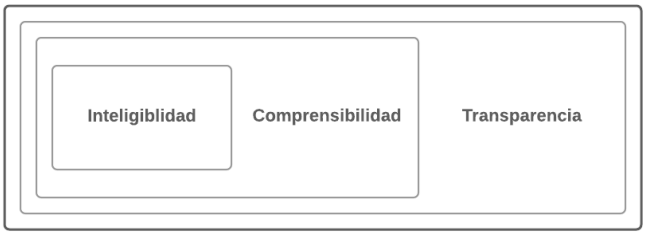


Figura 2. Gradación de los conceptos aplicables al modelo

En base a lo analizado para términos XAI relativos a modelos, la transparencia, con sus diferentes niveles (simulabilidad, descomponibilidad y transparencia algoritmica), representa un mayor nivel de entendimiento en XAI, seguida de la comprensibilidad, y por último de la inteligibilidad. En todo caso, en función del objetivo final del modelo de IA, será más útil considerar el modelo XAI desde la perspectiva ofrecida por uno u otro término. Por indicar un ejemplo, no siempre será necesario tratar el modelo desde la perspectiva de la transparencia algorítmica, en ocasiones será suficiente tratar el modelo para que el usuario confíe en que es comprensible.

III-B. Conceptos aplicables a los resultados del modelo

En este grupo estudiaremos los conceptos que son relativos a los resultados de un modelo. Estos conceptos son: explicabilidad, interpretabilidad y explicabilidad post-hoc.

Explicabilidad (explainability): Para este concepto revisamos las definiciones dadas por Adadi et al. [1], Montavon et al. [16] y Barredo et al. [3]. De forma generalizada lo que nos indican cada uno de ellos es que este concepto está asociado con la noción de explicación en la que el modelo presenta los resultados de forma que sea entendible tanto por el tomador de decisiones cómo para el humano que debe de entender el porqué de esa toma de decisión.

Bajo este punto hay que indicar que Adadi et al. en su estudio sobre este concepto [1] hace una observación sobre el número de veces que se hace referencia a la palabra explicabilidad frente a interpretabilidad en la comunidad *Machine Learning (ML)* e indica que aunque interpretable se utilice más veces que explicable, cuando se hace esta referencia, ambos conceptos se refieren al mismo significado. Por otra parte Montavon et al. [16] al igual que hacen en la definición dada para Comprensibilidad, vuelven a hablar del dominio interpretable en el que se da esa explicación hacia el tomador de decisiones.

Interpretabilidad (interpretability): La primera referencia que vamos a revisar es la dada por Doshi-Velez et al. [4] que también recogen Guidotti et al. [8] y Barredo et al. [3]. En estos estudios se indica que la interpretabilidad es la habilidad de explicar o presentar en términos comprensibles a un humano.

Otra referencia que encontramos de este concepto es la dada por Montavon et al. [16] en la que se indica que la interpretabilidad es el mapeo de un concepto abstracto a un dominio entendible para el humano. De acuerdo a esta definición, el resultado puede ser dado en diferentes contextos y formatos que se adapten al conocimiento de la persona que los revise.

Una vez revisadas tanto la definición de Interpretabilidad como la de Explicabilidad, proponemos la unificación de estos términos bajo el concepto de **Explicabilidad**. Basamos esta unificación de conceptos en los trabajos de Fernández et al. [8] y Mohseni et al. [1], dado que en el primero se realiza una equiparación de términos mientras que en el segundo se habla de una relación estrecha entre ambos.

Tras este análisis, entendemos que el término explicabilidad (equivalente a interpretabilidad más utilizado en la comunidad ML) hace referencia a la habilidad de un modelo de explicar la salida del modelo en términos comprensibles para el responsable de la toma de decisiones. Dicha explicación se puede realizar en diferentes formatos pero siempre teniendo en cuenta el dominio en el que el tomador de decisiones se encuentren.

Explicabilidad post-hoc (post-hoc explainability): Este concepto es un subtipo del término Explicabilidad revisado anteriormente. En la bibliografía especializada son dos los autores que nos hablan del mismo, Lipton [15] y Barredo et al. [3]. Nos indican que está dirigido a modelos que no son fácilmente interpretables debido a su diseño y de forma que no llegue a penalizar o sacrificar el rendimiento predictivo de los mismos.

Revisada la bibliografía especializada, proponemos como definición que la Explicabilidad Post-Hoc es aquella que nos permite poder interpretar y/o explicar modelos opacos o poco interpretables una vez estos han devuelto sus resultados sin que se penalice el rendimiento predictivo del mismo.

Un resumen de los conceptos analizados se muestra en Tabla I. En ella podemos ver para cada uno de estos términos la definición y agrupación propuesta así como las referencias a la bibliografía especializada que se ha analizado.

IV. Propiedades y objetivos de un modelo XAI

Una vez revisados los conceptos relacionados con XAI, analizaremos los diferentes términos relacionados con las propiedades u objetivos que ha de tener un modelo XAI. El análisis de estos términos en la bibliografía especializada permite comprobar que hay autores que etiquetan cada uno de estos términos de forma distinta aunque en su definición llegan a puntos comunes. Estos términos corresponden tanto a características deseables de los modelos de XAI como a objetivos que se pretenden alcanzar con estos modelos, que deberán incorporarse en sus mecanismos de evaluación.

Las propiedades y objetivos que se revisarán son: privacidad, fiabilidad, causalidad, transferibilidad, informatividad, imparcialidad e interactividad.

Privacidad (privacy): Hay dos autores en la bibliografía especializada que hablan de la privacidad, aunque refiriéndose a ella de forma distinta. Por un lado, Doshi-Velez et al. [4] habla de privacidad, mientras que Barredo et al. [3] hablan de la sensibilización o concienciación de la privacidad *(privacy awareness)*. Sin embargo, en ambos casos se indica que el modelo debe ser capaz de proteger en cada momento la información sensible que maneja, para evitar que el modelo sea entrenado por terceros no autorizados y se comprometa la privacidad de los datos.

A partir de estas definiciones, podemos establecer que, para que un modelo cumpla con la privacidad, debe ser capaz de proteger toda la información sensible que se maneje en cada una de sus etapas, independientemente de que el modelo sea entrenado por nosotros mismos o por terceros, siendo particularmente riguroso en las etapas de explicación del modelo para evitar exponer la privacidad de los datos sensibles al intentar hacer más explicable el modelo.

Fiabilidad (reliability): Distintos autores hablan de esta propiedad refiriéndose a ella con diferentes acepciones, que una vez definidas tienden a mostrar un significado muy similar. En el caso de Adadi et al. [1] utiliza los términos de fiabilidad *(reliability)* y robustez *(robustness)*, Lipton [15] habla del término confianza *(trust)* y Barredo et al. [3] utilizan los términos fiabilidad *(trustworthiness)* y confianza *(confidence)*, diferenciando entre ellos.

A partir de la revisión de las distintas definiciones encontradas en la bibliografía especializada, proponemos una definición unificada de este termino, de la siguiente forma: la fiabilidad es la seguridad y confianza de que un modelo actuará según lo previsto al enfrentarse a un problema determinado y funcionará correctamente con respecto a los objetivos y escenarios reales a los que se enfrenta, teniendo en cuenta que se puedan producir modificaciones en sus parámetros o variables de entrada.

Causalidad (causality): Esta propiedad se analiza por distintos autores en la bibliografía especializada: Barredo et al. [3], Lipton [15] y Adadi et al. [1]. Todos ellos se refieren a esta propiedad como causalidad, indicando de forma general que se pueden producir perturbaciones o inclusión de nuevas hipótesis durante el proceso que permiten encontrar relaciones causales dentro de los datos disponibles.

De acuerdo con esto, podemos indicar que la causalidad es la propiedad que permite que los modelos obtengan correlaciones de los datos de los que aprenden, pudiendo así proporcionar una primera intuición de posibles relaciones causales dentro de estos datos.

Transferibilidad (transferability): La transferibilidad es analizada utilizando este mismo nombre por Barredo et al. [3] y Lipton [15]. Ambos hacen referencia a esta propiedad como transferibilidad, *transferability*, indicando que la transferibilidad es la propiedad por la que un modelo ha de tener la capacidad de transmitir las habilidades aprendidas de forma que puedan ser reutilizadas en otros escenarios. En [3] se indica además que si un modelo es transferible no implica que sea explicable de por sí.

A partir de estas definiciones, proponemos definir la transferibilidad como la propiedad por la que un usuario puede aplicar de forma generalizada el conocimiento aprendido mediante un modelo a otros modelos que se dan en situaciones distintas.

Informatividad (informativeness): Para este término se han analizado las definiciones introducidas en Barredo et al. [3] y Lipton [15] , que utilizan el mismo nombre. En estos trabajos, se define la informatividad como el hecho de que un modelo es capaz de ampliar el conocimiento e información que tiene sobre el problema que se aborda, de forma que se puede relacionar la decisión tomada por el usuario con la solución dada por el modelo.

En vista de estas definiciones, proponemos definir la informatividad como el proceso por el cual el modelo debe transmitir información tanto acerca de sus resultados como del problema que se aborda para poder entender y relacionar la decisión tomada.

Tabla I
RESUMEN DE LOS CONCEPTOS APLICABLES A XAI

Tipo	Concepto	Definición	Referencias
Aplicables al modelo	Inteligibilidad	Un modelo es inteligible si funcionalmente es entendible por un ser humano sin necesidad de profundizar en el funcionamiento interno del mismo.	[3], [16], [15]
	Comprensibilidad	Un modelo es comprensible si tiene suficiente capacidad para representar de forma entendible para el ser humano el conocimiento aprendido sea cual sea la profundidad del mismo.	[3], [7], [8], [6]
	Transparencia	Un modelo es transparente cuando su modelo funcional es comprensible por sí mismo. Se definen distintos niveles de transparencia dependiendo del tipo de modelo de la siguiente forma. Si se trata de un modelo de caja negra, lo consideraremos transparente si sólo su algoritmo de entrenamiento es comprensible funcionalmente (transparencia algorítmica). En el caso de un modelo de caja blanca, para considerar que un modelo es transparente, debe de ser comprensible funcionalmente a nivel de modelo completo (simulabilidad), es decir, debe de ser comprensible funcionalmente de forma modular a partir de su descomposición a la vez que también lo es su algoritmo de entrenamiento.	[3], [15]
Aplicables a los resultados	Explicabilidad	Habilidad de un modelo de explicar la salida del modelo en términos comprensibles para el responsable de la toma de decisiones. Dicha explicación se puede realizar en diferentes formatos pero siempre teniendo en cuenta el dominio en el que el tomador de decisiones se encuentren.	[3], [16], [1]
	Explicabilidad Post-Hoc	Es aquella que nos permite poder interpretar y/o explicar modelos opacos o poco interpretables una vez estos han devuelto sus resultados sin que se penalice el rendimiento predictivo del mismo.	[3], [15]

Imparcialidad (fairness): A este término hacen referencia distintos trabajos bajo distintas formas. Barredo et al. [3] y Mosheni et al. [4] utilizan el término imparcialidad, definiéndolo como la capacidad de garantizar la imparcialidad en los modelos permitiendo que los grupos protegidos no sean discriminados de forma que se puedan eliminar los sesgos en los datos en los que se expone el modelo. Por su parte, Lipton [15] habla de la toma de decisiones justa y ética *(fair and ethical decision-making)*, y como cada vez se está haciendo más presente la preocupación de la evaluación e interpretación de las decisiones producidas automáticamente para verificar que las mismas se ajustan a las normas éticas.

Teniendo en cuenta las distintas definiciones, proponemos definir la imparcialidad como la propiedad por la cual los modelos de inteligencia artificial explicable deben garantizar una toma de decisiones justa y ética cuando se decide un resultado a partir de la salida del modelo, de la misma forma que los datos utilizados, tanto en el entrenamiento como durante la ejecución del modelo, no debiendo estar perjudicados por ningún tipo de sesgo, ya sea en datos, algoritmos u otros.

Interactividad (interactivity): En Barredo et al. [3] se hace referencia a la interactividad indicando que uno de los objetivos de un modelo de ML y por ende de la IA, es que sea interactivo para el usuario. ¿Cómo se puede conseguir un modelo que sea interactivo para el usuario? Montavon et al. [16] indican que para ello es necesario que el modelo se realice bajo un dominio que sea interpretable para el tomador de decisiones.

De acuerdo con esto, proponemos definir la interactividad como la forma en la que un modelo es capaz de transmitir de forma simple y comunicativa su resultado, ya sea a través de visualizaciones gráficas, lenguaje natural, etc. Para ello, ha de hacerlo bajo el dominio interpretable del usuario.

La Tabla II resume estas propiedades y objetivos. En ella podemos ver para cada uno de estos términos la definición propuesta así como las referencias a la bibliografía especializada que se ha analizado.

V. CONCLUSIONES

En la bibliografía especializada en XAI no existe un consenso respecto a la denominación y definición de los conceptos utilizados en el área. En este trabajo se describe parte de los resultados del análisis de la bibliografía existente con el objetivo de extraer información y fomentar el debate en el área que permita llegar a una terminología unificada. Debido a la gran importancia que día a día está cobrando XAI, creemos importante realizar una unificación de esta terminología y poder sentar las bases en las que seguir trabajando.

En este estudio, se ha realizado un análisis exhaustivo de las referencias encontradas en la bibliografía especializada sobre XAI, que ha permitido diferenciar y unificar -según el caso- conceptos, proponer definiciones integradoras y avanzar en el conocimiento sobre propiedades de un modelo XAI que serán objetivos a considerar en su diseño y/o evaluación.

Es mucho lo que queda por hacer en esta rama de la IA, y una de las líneas de trabajo en las que seguiremos profundizando será en intentar aproximar cuán explicable es un modelo. Ya encontramos autores que describen un modelo de medición, como es el caso de Mosheni et al. [17] y Hoffman et al. [11]. Este modelo cuantifica en base a encuestas y/o entrevistas, al tomador de decisiones, la explicabilidad del modelo. Es por esto que entendemos que en XAI habrá dos tipos de métricas, las métricas subjetivas que irán orientadas a la cuantificación de la explicabilidad del modelo a partir del entendimiento del tomador de decisiones, como se indica

Tabla II
RESUMEN DE LAS PROPIEDADES Y OBJETIVOS APLICABLES A XAI

Característica	Definición	Referencias
Privacidad	Para que un modelo cumpla la propiedad de privacidad, este debe de ser capaz de proteger toda la información sensible de un individuo que maneje en cada una de sus etapas, ya sea entrenado por nosotros mismos o por terceros, siendo particularmente riguroso en las etapas de explicación del modelo, ya que no podemos exponer la privacidad de los datos sensible por hacer más explicable el modelo.	[4], [3]
Fiabilidad	Es la seguridad y confianza de que un modelo actuará según lo previsto al enfrentarse a un problema determinado y funcionará correctamente con respecto a los objetivos y escenarios reales a los que se enfrenta teniendo en cuenta que se puedan producir modificaciones en su parámetros o variables de entradas.	[3], [15], [1]
Causalidad	Es aquella propiedad que permite obtener correlaciones de los datos que aprende de forma que pueda proporcionar una primera intuición de posibles relaciones causales dentro de estos datos.	[3], [15], [1]
Transferibilidad	Es la propiedad por la cual un usuario puede aplicar de forma generalizada el conocimiento aprendido durante un modelo a modelos que se dan en situaciones distintas.	[3], [15]
Informatividad	Es el proceso por el cual el modelo debe de transmitir información acerca de sus resultados como del problema que se aborda para poder entender y relacionar la decisión tomada.	[3], [15]
Imparcialidad	Es el objetivo por el cual, los modelos de inteligencia artificial explicable deben garantizar una toma de decisiones justa y ética cuando se decide un resultado a partir de la salida del modelo, de la misma forma que los datos utilizados, tanto en el entrenamiento como durante la ejecución del modelo, deben de no estar perjudicados por ningún tipo de sesgo, ya sea en datos, algoritmos u otros.	[3], [15], [4]
Interactividad	Forma en la que un modelo es capaz de transmitir de forma simple y comunicativa su resultado, ya sea a través de visualizaciones gráficas, lenguaje natural, etc. Para ello, ha de hacerlo bajo el dominio interpretable del usuario.	[3], [16]

en [17] y [11], y las métricas objetivas, que cuantifican la medición de explicabilidad del modelo a partir de las propiedades del mismo. El análisis de este último tipo de métricas será la línea de trabajo en la que seguiremos investigando. Otra de las líneas de trabajo que se nos abre es el desarrollo de una taxonomía más formal que pueda ser aceptada por la comunidad científica.

RECONOCIMIENTOS

Este trabajo ha sido subvencionado por el Ministerio de Ciencia, Innovacion y Universidades bajo el proyecto PID2019-107793GB-I00 (Fondos FEDER).

REFERENCIAS

[1] A. Adadi and M. Berrada, “Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI),” IEEE Access, (6) (2018) 52138-52160, 2018, https://doi.org/10.1109/ACCESS.2018.2870052

[2] S. Barocas, S. Friedler, M. Hardt, J. Kroll, S. Venka-Tasubramanian, and H. Wallach. “The FAT-ML Workshop Series on Fairness, Accountability and Transparency in Machine Learning.” Accessed: Jun. 6, 2018

[3] A. Barredo, N. Díaz-Rodriguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. García, S. Gil-López, D. Molina, R. Benjamins, R. Chatila, F. Herrera, “Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI,” Information Fusion (58) (2020) 82-115, https://doi.org/10.1016/j.inffus.2019.12.012

[4] F. Doshi-Velez and B. Kim,“Towards a rigorous science of interpretable machine learning,” 2017, [Online]. Available: [Online]. Available: arXiv:1702.08608v2

[5] M. Dye, C. Ekanadham, A. Saluja, A. Rastogi, “Supporting content decision makers with machine learning”, 2020, [Online]. Available: https://netflixtechblog.com/supporting-content-decision-makers-with-machine-learning-995b7b76006f. Accessed on: May. 20, 2021

[6] A. Fernandez , F. Herrera , O. Cordon , M. Jose del Jesus , F. Marcelloni , “Evolutionary fuzzy systems for explainable artificial intelligence: Why, when, what for, and where to?”, IEEE Computational Intelligence Magazine 14 (1) (2019) 69-81

[7] A. A. Freitas, “Comprehensible Classification Models: A Position Paper”. Association for Computing Machinery (15, 1) (2014) 1-10, https://doi.org/10.1145/2594473.2594475

[8] R. Guidotti , A. Monreale , S. Ruggieri , F. Turini , F. Giannotti , D. Pedreschi , “A Survey of Methods for Explaining Black Box Models”, ACM Comput. Surv. (51) (2018), https://doi.org/10.1145/3236009

[9] D. Gunning, DARPA, “Explainable Artificial Intelligence (XAI)”, IJCAI-16 DLAI WS, 2016

[10] D. Gunning. “Explainable artificial intelligence (XAI), Defense Advanced Research Projects Agency (DARPA)”. Accessed: Jun. 6, 2018

[11] R. R Hoffman, S. T. Mueeler, G. Klein, J. Litman, “Metrics for Explainable AI: Challenges and Prospects,” 2019. arXiv:1812.04608v2

[12] A.M. Ismael, A. Sengür, “Deep learning approaches for COVID-19 detection based on chest X-ray images”, Expert Systems with Applications (164) 2021, https://doi.org/10.1016/j.eswa.2020.114054

[13] W. James Murdoch, C. Singh, K. Kumbier, R. Abbasi-Asl y B. Yu, “Definitions, methods, and applications in interpretable machine learning”. Proceedings of the National Academy of Sciences (44) (2019) 22071-22080, http://dx.doi.org/10.1073/pnas.1900654116

[14] M. van Lent, W. Fisher, M. Mancuso, “An Explainable Artificial Intelligence System for Small-unit Tactical Behavior”. 16th Conf. Innov. Appl. Artif. Intell., 2004

[15] Z. C. Lipton, “The Mythos of Model Interpretability: In Machine Learning, the Concept of Interpretability is Both Important and Slippery.,” Queue (16) (2018) 31-57, https://doi.org/10.1145/3236386.3241340

[16] G. Montavon, W. Samek, K.-R. Müller, “Methods for interpreting and understanding deep neural networks” Digital Signal Processing (73) (2018) 1-15, https://doi.org/10.1016/j.dsp.2017.10.011

[17] S. Mosheni, N. Zarei y E.D. Ragan, “A Multidisciplinary Survey and Framework for Design and Evaluation of Explainable AI Systems”, 2020. [Online]. Available: arXiv:1811.11839v5

[18] G. Vilone y L. Long, “Explainable Artificial Intelligence: a Systematic Review”, 2020. [Online]. Available: arXiv:2006.00093v4

Challenges of Definition Extraction in Spanish Legal Texts

Karen Vázquez-Flores
Ontology Engineering Group
Universidad Politécnica de Madrid
kvazquez@delicias.dia.fi.upm.es

Patricia Martín-Chozas
Ontology Engineering Group
Universidad Politécnica de Madrid
pmchozas@fi.upm.es

Elena Montiel-Ponsoda
Ontology Engineering Group
Universidad Politécnica de Madrid
emontiel@fi.upm.es

***Abstract*—In this paper, we describe a preliminary experiment focused on definition extraction over Spanish legal texts based on a number of linguistic patterns and supported by an automatic term extraction tool. Through this experiment, we analyse current issues on the definition extraction task in general and, specifically, open challenges on the automatic identification of legal definitions in Spanish. As a result of this analysis, we also suggest various ideas on possible solutions to those issues and next steps in our work.**

***Index Terms*—Definition Extraction, Legal Texts, Terminology**

I. Introduction

Information Extraction (IE) and Information Retrieval (IR) are two areas of research in the Natural Language Processing (NLP) field whose aim is to obtain information from unstructured texts. Depending on the type of information to be identified, (be it names of companies or people mentioned in the text, terms dealt with in the document, or opinions given in Tweets), we will be referring to Named Entity Recognition, Term Extraction, or Opinion Mining, to mention but a few of the more specific tasks encompassed by these areas. When dealing with Term Extraction, the aim is to identify those terms or keywords that better define the content of a document. Terms may provide a network of superficial clues to understand whether a document fits the expected search criteria [5]. Automatic Term Acquisition may also be used for indexing new documents, or automatically annotating documents with the resulting terminology. In this sense, the richer the information describing the term, the more accurate the annotations will be.

With the purpose of adding definitions to terms extracted from a legal corpus, we have implemented a set of patterns to automatically discover definitions of Spanish legal terms implicitly defined in texts. It is well recognised that Legal language is a professional jargon, and, as such, it has some particularities not so frequently found in plain language. Spanish legal language is characterised by the frequent use of technical terms, archaisms, and complex syntactic constructions (abundant periphrasis, long subordinate clauses, gerunds, impersonal and passive verbs, long enumerations, use of future subjunctive and future tense for obligation), preference for the 3rd person singular, aphoristic style, or the overuse of citations and references.

Definitions are also abundant in legal documents. Many terms will be directly defined in the text the first time they are mentioned. Some legal documents include a glossary of terms and definitions at the end of the document or as an annex (for example, European Directives and Regulations usually devote one article to define the main terms in the document[1]). However, this is an uncommon practice in Spanish national law, and discovering definitions in text is a necessary and challenging task. The way in which legal terms are defined in legal discourse follows certain lexico-syntactic patterns that are not common in plain language.

In order to palliate the lack of approaches to automatically retrieve definitions of legal terms in Spanish legal documents, in this contribution we present an approach that strongly relies on lexico-syntactic patterns, understood here as a combination of lexical items following a certain syntactical structure that is recursively used to express a relation of interest, a definitional pattern in this case. Apart from the identification and implementation of those patterns, we have also worked in a previous automatic term extraction process over the corpus. In this regard, our objective was to test if this process improves the performance of the pattern extraction.

The rest of the document is structured as follows: firstly, we briefly refer to some works on the extraction of definitions in legal texts. Then, we refer to the main features of definitions in law, and describe the patterns we have created and their implementation. Finally, we discuss the results obtained with and without the help

[1] https://eur-lex.europa.eu/eli/reg/2016/679/oj

of a terminology extraction tool, and refer to future lines of work.

II. Related work

The most commonly applied approaches to definition extraction are pattern based approaches. Their main drawback lays on the consuming process involved in the knowledge elicitation task; however, correctly built, patterns usually retrieve accurate results, as tested in various previous works. For instance, in 2004, Saggion applied a method that built the basis for a definition-based question-answering system [9]. The reproducibility of this work was, however, limited to a restricted number of definitions. In the case of [4], the authors propose patterns based on verbs and verbal phrases in Romanian, being the most productive ones those in which definitions were introduced by *denote*, *state*, *represent*, *define*, *specify*, *consist*, *name* and *permit*. A similar work, with similar results, is performed in [8] to retrieve definitions for Slavic languages. There have also been pattern-based approaches specialised in the legal domain, such as [10], in which the authors identify 52 rules based on lemma/POS and dependency parsing to extract definitions from a large corpus of German court decisions. To palliate the low recall they obtained, they used bootstraping techniques based on single seed words at the expense of precision. The authors suggest that a balance between precision and recall is to be further achieved.

Leaving pattern- and rule-based approaches aside, there have also been several experiments relying on Machine Learning. The scalability of those works is always much higher that the one achieved with pattern-based approaches, but the manual work required for the annotation of a reference corpora is highly expensive. This is the case of [1], where they explored the use of machine learning to extract definitions from non-technical English texts, or [2], where results suggest that naive Bayes and random forest have the most suited learning bias for the task of definition extraction.

Mixed techniques have likewise been explored to identify definitions with higher linguistic variability than the classic encyclopedic *genus-et-differentia* definition. In 2009, Westerhout [11] proposed a definition extraction method for the automatic creation of glossaries within the area of eLearning. They used a sequential combination of a rule-based approach, based on POS-tags and a Machine Learning classification algorithm based on Balanced Random Forest. Another mixed work is [3], in which a weakly supervised bootstrapping approach was used to classify sentences in Wikidata as being definitions or not. In general, this kind of hybrid approaches show competitive results in comparison with others only based on patterns, but there is no evidence of works in specialised languages or languages other than English.

All in all, most of the reviewed works suggest that identifying definitions is a challenging task that is dependant on the language and the area of expertise, and that probably mixed approaches are expected to provide effective solutions. Little work has been done with regard to legal definition extraction in Spanish. Our aim is to take the first steps in this topic through a preliminary experiment based on patterns and helped by an automatic terminology extraction tool. We analyse the main issues and challenges derived from this experiment, propose ideas to solve them and expose future steps.

III. Definitions in law

A *definition* is a piece of text that helps clarifying the specific sense of an expression that appears several times throughout a document [7]. Definitions are divided into two parts: *definiendum* and *definiens*. The *definiendum* is the word or element that is to be defined, and the *definiens* is the sentence or clause that gives the meaning to the *definiendum*. For example, in the definition "semantic is the science of meaning", the *definiendum* is "semantic" while the *definiens* is "the science of meaning". These parts are usually joined by a *copula*, which, in this case, would be "is". The most common structure of definitions is, hence, *definiendum*, *copula* and *definiens*.

However, these components are not always present in a fixed order. Specially in legal texts, it is common that the *definiens* component appears before the *definiendum*, and that the *copula* is not always expressed with the verb to be, which can sometimes lead to misunderstandings.

Specifically in Spanish legal texts, it is natural to find the structure *copula* + *definiendum* + *definiens*, for instance, "son documentos públicos los autorizados por un notario [...]" (*are public documents those authorised by a public notary**), where the *copula* is "son", the *definiendum* is "documentos públicos" and the *definiens* is "los autorizados por un notario". The *copula* can also be expressed with other verbs, commonly pronominal structures, as for instance "se considera" (*it is considered as*), "se entiende por" (*it is understood as*).

Notwithstanding, and depending on the legislator, other forms can also prevail in a text. For instance: "there is/exists" + *definiendum* + a condition expressed by "when", "if", "if and only if", or similar. In article 392 of the Spanish Civil Code, we can find the following definition: "hay comunidad cuando la propiedad

de una cosa pertenece a varias personas" (*there exists a community when the property of something belongs to several people*). And it is also not uncommon to find definitions introduced by some punctuation marks, that is, *Definiendum: Definiens*, or in apposition to the *Definiendum* between commas.

IV. Experiment

As theoretical foundation, we rely on the study of definitions in Spanish legal texts by [7]. Taking into account the different types of definitions that can be found in a legal text, we have identified the most standard patterns for legal definitions in Spanish (see Table I).

Our object of study is one of the most representative texts of Spanish labour law, the Spanish Workers' Statute[2]. As a first step when applying linguistic patterns, we perform a part-of-speech tagging of the text. For that purpose, in this contribution we have used CoreNLP[3] and spaCy[4].

To identify the *definiendum*, we have followed two different strategies: 1) the one self generated by POS tokens and 2) the one assisted by an automatic term extraction tool. In the first case, we have generated sequences of lexico-syntactic patterns based on each token's part of speech (see Table II). In the second case, we have made use SketchEngine[5] [6], a widely used corpora managing tool that also implements cutting-edge automatic term extraction algorithms. Our hypothesis is that by relying on the identification of the most relevant terms in the text, we may reduce the number of false positive results obtained, which identify as *definiendum* structures that do not correspond to any legal term.

After executing the two different approaches, we then compare the results obtained to check if previous terminology work can help in definition extraction, or if, on the contrary, identifying terms with part-of-speech tags works better in this specific case.

V. Results and Discussion

The main issue common to both approaches is, as expected, the great number of false positives retrieved. However, this study confirms the assumption that when relying on previously extracted terms, the number of false positives decreases. Still, in each approach we have been faced with different issues.

As for the definitions extracted with lexico-syntactic patterns, the major problem is caused by the erroneously assigned POS-tags. Lexico-syntactic patters rely completely on the accurate tagging of words in a text. Regarding those definitions retrieved after a previous term extraction task, we also rely on the accurate performance of the automatic term extraction tool, which may also wrongly identify tokes as terms. For instance, we retrieved the following definition: *A los anteriores efectos, se considerará salario la cantidad reconocida como tal en acto de conciliación o en resolución judicial por todos los conceptos a que se refiere el artículo 26* (For the above mentioned purposes, shall be considered salary the amount recognized as such in an act of conciliation or in a judicial resolution for all the concepts referred to in article 26*). The term defined in this case is "salario" (salary). However, the term extraction tool wrongly identified "salario la cantidad" (salary the amount) as the multi-word term being defined.

Similarly, we also retrieved: *El contrato de trabajo, se entenderá celebrado a tiempo parcial cuando se haya acordado la prestación de servicios durante un número de horas al día, a la semana, al mes o al año, inferior a la jornada de trabajo de un trabajador a tiempo completo comparable.* (*The employment contract shall be understood as part-time when the provision of services is agreed for a number of hours per day, per week, per month or per year, lower than the working day of a comparable full-time worker*). The *definiendum* in this case is "celebrado a tiempo parcial"; however, this term does not appear in our term list, and our patterns wrongly matched this definition with the term "tiempo parcial", that has a broader scope.. Therefore, the automatically extracted terms need to be post-processed, either by humans or with an automatic post-processing approach to avoid this kind of mistakes.

A different issue occurs when we find part of the term before the copula and part of the term after the copula. For instance, in this defintion of the term fair dismissal, we have the term "despido" (dismissal) before the copula, and the modifying adjective "procedente" (fair) after the copula. See: *El despido se considerará procedente cuando quede acreditado el incumplimiento alegado por el empresario en su escrito de comunicación* (The dismissal will be considered fair when the breach alleged by the employer in his communication letter is proven*). A human can reason that the term defined is "despido procedente", but our algorithm cannot. Therefore, in this sentence, we have different term choices: *How can a machine reason which of the terms is being defined?* This is doubtlessly a very interesting challenge to work on.

To evaluate the performance of the patterns, with and without previous term extraction, we have created a gold standard by manually annotating the definitions

[2]https://www.boe.es/eli/es/rdlg/2015/10/23/2/con
[3]https://stanfordnlp.github.io/CoreNLP/
[4]https://spacy.io/
[5]https://www.sketchengine.eu/

TABLE I
Standard patterns with examples. *We have included literal translations in order to show the bilingual examples of the patterns.

	Spanish Patterns (and English translation)	Examples in English and Spanish
1	Hay/Existe + definiendum + cuando/siempre/if + definiens (There is/exists + definiendum + when/if + definiens)	*There is community when there are more than five people.** *Existe una comunidad cuando hay más de 5 personas**
1	Se considera + definiendum + definiens (It is considered + definiendum + definiens)	*It is considered collective agreement an agreement that...* *Se considera convenio colectivo un convenio que...*
3	Definiendum + es/son + definiens (Definiendum + is/are + definiens)	*Workers are people who work.* *Trabajadores son aquellas personas que trabajan.*
4	Es/son + definiendum + definiens (It/They is/are + definiendum + definiens)	*It is a worker the person who works.** *Es un trabajador la persona que trabaja.**
5	Se define + definiendum + como + definiens (It is defined + definiendum + as + definiens)	*It is defined worker as a person who works.** *Se define trabajador como una persona que trabaja.**
6	Se define + como + definiendum + definiens (It is defined + as + definiendum + definiens)	*It is defined as worker a person who works.** *Es definido como trabajador una persona que trabaja.**
7	Se entiende que + verb + definiendum + definiens (It is understood that + verb + definiendum + definiens)	*It is understood that there is a collective contract when...** *Se entiende que hay contrato colectivo cuando...**

TABLE II
Term patterns with approximate translations in English

	Term patterns	Term examples (with approximate English translation)
1	haber + noun	ley (law)
2	denominar + noun + conj + noun	empleador o empresario(employer or businessman)
3	considerarán + noun	familiares (familiar)
4	considerarán + noun + adj	relaciones laborales (labour relations)
5	considerará + noun + adp + noun	centro de trabajo (workplace)
6	considerará + adp + noun + adj	de carácter colectivo (collective character)
7	considerará + verb + det + noun	terminado el contrato (finished contract)
8	entender + adj + adv	prorroga automática (automatic extension)
9	entender + noun + adj	grupo profesional (professional group)
10	entender + noun + adp + noun	reducción de jornada (reduction of working hours)
11	entender + verb + adp + det + noun + adj	excluida de el ámbito laboral (excluded from the workplace)
12	entender + adj + adp + noun + adj	celebrado por tiempo indefinido (for indefinite period)
13	entender + adj + conj + adp + noun	nulos y sin efectos (null and void)
14	entender + verb + noun + adj	causas técnicas (technical reasons)
15	entender + verb + noun + adp + det + noun	insolvencia del empresario (insolvency of the employer)
16	entender + det + noun + aux + adj	la disminución es persistente (persistent decrease)
17	denominar + propn + adp + propn	código de trabajo (work code)
18	denominarán + noun	empresario (entrepreneur)
19	denominarán + noun + adj + adp + det + adj	personalidad jurídica de el contratante (legal entity of the contractor)
20	denominarán + noun + adp + det + noun	trabajadores de el contratista (workers of the contractor)
21	denominarán + noun + adp + noun	semanas de disfrute (weeks of leave)
22	denominarán + punce + det + noun + conj	el contratista o subcontratista (contractor or subcontractor)

contained in the Spanish Workers' Statute. Results are exposed in table III The annotations were made by three linguist professionals, and they are available amongst the rest of the material involved in the experiment in our Github repository[6].

We have also compared the automatically identified definitions with the ones provided in the most important source of legal lexicographic knowledge in Spain: Pan-Hispanic Dictionary of Legal Spanish (*Diccionario Panhispánico del Español Jurídico, DPEJ* henceforth)[7]. Additionally, we also note that the definitions of some terms in the dictionary slightly differ from the ones we provided in the Workers' Statute. In this regard, we assume that terms may be more specifically defined in the context of a certain legislation, and that we may find various definitions for the same term in different norms. We plan to further explore this issue. Examples of terms that are not included in the DPEJ are:

TABLE III
In this table we compare the number of definitions manually annotated in the gold standard and the number of definitions identified by the patterns, without and with a previous term extraction stage.

Gold Standard	Without term extraction	With term extraction
66 annotated definitions	25 true positives	34 true positives
	11 false positices	3 false positives

[6]https://github.com/krnlet/legal_definition_extraction
[7]https://dpej.rae.es/

- Trabajador nocturno (night worker): *Para la aplicación de lo dispuesto en el párrafo anterior, se considerará trabajador nocturno a aquel que realice normalmente en periodo nocturno una parte no inferior a tres horas de su jornada diaria de trabajo, así como a aquel que se prevea que puede realizar en tal periodo una parte no inferior a un tercio de su jornada de trabajo anual.* (For the application of the provisions in the previous paragraph, it is considered a night worker who normally performs at least three working hours per day or a third of their working hours per year during the night.)
- Familiar (family member or relative): *Se considerarán familiares, a estos efectos, siempre que convivan con el empresario, el cónyuge, los descendientes, ascendientes y demás parientes por consanguinidad o afinidad, hasta el segundo grado inclusive y, en su caso, por adopción.* (For these purposes, they will be considered family members, provided they live with the employer, the spouse, the descendants, ascendants and other relatives by consanguinity or affinity, up to and including the second degree and, where appropriate, by adoption.)
- Código de Trabajo (Employment Code): *El Gobierno, a propuesta del Ministerio de Empleo y Seguridad Social, recogerá en un texto único denominado Código de Trabajo, las distintas leyes orgánicas y ordinarias que, junto con la presente, regulan las materias laborales, ordenándolas en títulos separados, uno por ley, con numeración correlativa, respetando íntegramente su texto literal.* (The Government, at the proposal of the Ministry of Employment and Social Security, will collect in a single text called the Employment Code, the different organic and ordinary laws that, together with the present, regulate labour matters, ordering them in separate titles, one by law, with consecutive numbering, fully respecting its text.)

An additional advantage of performing definition extraction over legal texts is that we can identify when the same term has been defined differently in several legal documents, for comparison purposes, and also, when several (complementary) definitions are available for the same term in the same document, as is the case of "centro de trabajo", work centre, in the document at hand:

1) *A efectos de esta ley se considera centro de trabajo la unidad productiva con organización específica, que sea dada de alta, como tal, ante la autoridad laboral.*(For the purposes of this law, the productive unit with a specific organization, which is registered, as such, before the labor authority is considered a work center).
2) *En la actividad de trabajo en el mar se considerará como centro de trabajo el buque, entendiéndose situado en la provincia donde radique su puerto de base.* (In the activity of work at sea, the ship will be considered as a work center, understood to be located in the province where its base port is located).

This kind of "contextual enrichment" seems highly valuable, since lexicographic resources such as the DPEJ usually contain only one and more generic definition.

VI. Conclusions and Future Work

In this experiment, we have tackled the case of standard definitions in Spanish legal texts. Our approach relies on POS-taggers that have shown imperfect for Spanish. Research on more accurate taggers and lemmatizers is therefore a requirement. Other issues have been thrown by our own lexico-syntactic patterns in combination with the terms previously identified, which sometimes generated wrong results and also need to be reviewed and adjusted for the domain at hand. With regard to the automatic term extraction tool applied, SketchEngine, we observed that several of the terms identified are not completely correct nor relevant in the text. Thus, another step to take is to test the performance of different automatic term extraction tools, which may apply statistical or linguistic approaches, and also to research on a post-processing algorithm to remove noise from those terms. A curated term list would surely improve the definitions extraction process.

On the other hand, in this experiment we have left aside the so-called *non-standard legal definitions*. Such definitions can be divided into four different groups: 1) indirect definitions, 2) conditional definitions, 3) incidental definitions and 4) meta-linguistic definitions. The first group includes regular definitions but expressed in a more indirect or verbose manner. The second group refers to definitions that have to comply with more than one condition: if P, then X is A only if X is B. Incidental definitions are those that have been exposed without intention, while exposing another statement. Finally, meta-linguistic definitions are those in which the *definiendum* can be both *definiendum* and *definiens*. Examples on these types of definitions can be found in [7].

It goes without saying that such definitions are much more difficult to extract than the standard ones. In fact, we have also tried to apply lexico-syntactic patterns for this purpose with little success (for the moment).

A clear future step is to work on the automatic identification of non-standard definitions in legal texts. A preliminary idea would involve relying on the manual annotation of non-standard definitions of the Spanish Workers' Statue to train a machine/deep learning engine to recognise them. However, we do not completely discard the use of linguistic patterns as well. A combination of both could also turn into an interesting and effective approach.

Acknowledgements

This work has received funding from the EU's Horizon 2020 Research and Innovation programme through the contracts Marie Sklodowska-Curie PROTECT (grant agreement No. 813497), Lynx (grant agreement No. 780602) and Prêt-à-LLOD (grant agreement No. 825182), and from the Spanish Ministry of Economy, Industry and Competitiveness through the Datos4.0 contract (TIN2016-78011-C4-4-R).

References

[1] C. Borg, M. Rosner, and G. Pace. Evolutionary algorithms for definition extraction. In *Proceedings of the 1st Workshop on Definition Extraction*, pages 26–32, 2009.

[2] R. Del Gaudio, G. Batista, and A. Branco. Coping with highly imbalanced datasets: A case study with definition extraction in a multilingual setting. *Natural Language Engineering*, 20(3):327–359, 2014.

[3] L. Espinosa-Anke, F. Ronzano, and H. Saggion. Weakly supervised definition extraction. In *Angelova G, Bontcheva K, Mitkov R, editors. International Conference on Recent Advances in Natural Language Processing 2015 (RANLP 2015); 2015 Sept 7-9; Hissar, Bulgaria. Stroudsburg: ACL (Association for Computational Linguistics); 2015. p. 176-85.* ACL (Association for Computational Linguistics), 2015.

[4] A. Iftene, D. Trandabăţ, and I. Pistol. Grammar-based automatic extraction of definitions and applications for romanian. In *Proceedings of RANLP workshop" Natural Language Processing and Knowledge Representation for eLearning environments*, pages 19–25, 2007.

[5] C. Jacquemin. *Spotting and discovering terms through natural language processing*. MIT press, 2001.

[6] A. Kilgarriff, V. Baisa, J. Bušta, M. Jakubíček, V. Kovář, J. Michelfeit, P. Rychlỳ, and V. Suchomel. The sketch engine: ten years on. *Lexicography*, 1(1):7–36, 2014.

[7] R. L. H. Marín. Definiciones en el derecho. *Anuario de filosofía del derecho*, (11):367–380, 1994.

[8] A. Przepiórkowski, Ł. Degórski, M. Spousta, K. Simov, P. Osenova, L. Lemnitzer, V. Kubon, and B. Wójtowicz. Towards the automatic extraction of definitions in slavic. In *Proceedings of the Workshop on Balto-Slavonic Natural Language Processing*, pages 43–50, 2007.

[9] H. Saggion. Identifying definitions in text collections for question answering. In *LREC*, 2004.

[10] S. Walter. Linguistic description and automatic extraction of definitions from german court decisions. In *LREC*, 2008.

[11] E. Westerhout. Definition extraction using linguistic and structural features. In *Proceedings of the 1st Workshop on Definition Extraction*, pages 61–67, 2009.

Bilingual Dataset for Information Retrieval and Question Answering over the Spanish Workers Statute

Pablo Calleja
Ontology Engineering Group
Universidad Politécnica de Madrid
pcalleja@fi.upm.es

Patricia Martín Chozas
Ontology Engineering Group
Universidad Politécnica de Madrid
pmchozas@fi.upm.es

Elena Montiel-Ponsoda
Ontology Engineering Group
Universidad Politécnica de Madrid
emontiel@fi.upm.es

Víctor Rodríguez-Doncel
Ontology Engineering Group
Universidad Politécnica de Madrid
vrodriguez@fi.upm.es

Elsa Gómez
Cuatrecasas

Pascual Boil
Cuatrecasas

***Abstract*—A bilingual dataset of questions and answers over a key document in Spanish labor law legislation, the Workers Statute, is presented. The document contains 150 questions and their respective answers in the form of one part number from the 130 parts in which the Workers Statute is divided (articles and other provisions), and with the most relevant excerpt of information for the answer. A simple system to retrieve the answers using standard technologies is also described, providing baseline numbers for accuracy in this task. This dataset may be of interest for researchers in the area of Q&A over legal documents. Both the question and answers are available in English and Spanish languages.**

***Index Terms*—Dataset, Q&A, Information Retrieval, Labor Law, Knowledge Graph**

I. Introduction

Information Retrieval and Question Answering have become core tasks in the so called knowledge-based society we live in. Search engines are our best allies when searching for information, regardless of topic or level of expertise. As described by Manning [5], Information Retrieval (IR) used to be an activity that only a few people engaged in such as reference librarians, paralegals, and similar professional searchers. Now the world has changed, and hundreds of millions of people engage in information retrieval when they use a web search engine.

In the legal domain, Information Retrieval and Question Answering take a substantial part in the daily work of lawyers when performing domain-specific searches due to the large amounts of text-based information generated in this area. Such a specific area relies heavily on the use of the appropriate terms and the relations between them as established in the search queries.

However, research improvements could not be possible if there are not evaluations to measure the performance of the new techniques, algorithms and systems. Particularly in the legal domain, there are a lot of scenarios in which particular use cases have to be achieved. Moreover, legal information is subject to constant changes and updates, and, in the European context, dependant of each jurisdiction.

In this paper, a new dataset for Information Retrieval and Question Answering has been created in collaboration with domain experts from the Spanish law-firm Cuatrecasas. The dataset is comprised of 150 labor law questions related to the Spanish Workers Statute. The information in the document is divided into Titles, Chapters, Sections, and Articles. Each question in the dataset is matched to the corresponding article in the Workers Statute in which the answer is contained, and also to the specific excerpt of the article in which the question is answered. The questions, articles and paragraphs are available in Spanish and English languages. In addition, the segmented Workers Statute is distributed along with the dataset.

Moreover, this paper presents an initial experiment in the context of the European Project Lynx in which the dataset is used for evaluation purposes.

This paper is structured as follows: In Section II we briefly refer to similar initiatives. In Section III the

This work has been supported by the Lynx project, which has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 780602.

dataset is described, and Section IV is devoted to the experiment. Finally, Section V presents the results of preliminary evaluations performed on the datasets, and conclusions and future steps are reflected in Section VI.

II. Related Work

Most of the datasets for Information Retrieval and Question Answering in the legal domain are oriented to exploit court cases in English language [8] which are also distributed in public research Tasks [3], [4]. For Spanish language, the recent publication of Legal-ES [7] poses a large scale Spanish corpus comprised of documents from different sources. However, the Spanish Workers Statute is not reflected in the corpus.

The value of this paper is to create a dataset focused on Spanish language for labor law questions related to the Workers Statute document.

III. Dataset

The dataset has been created during the developing of the European Project Lynx. Cuatrecasas, as part of the Lynx consortium, is interested in Information Retrieval systems over documents related to the Spanish legislation. The concrete use case is focused on the Spanish Workers Statute.

The dataset has been created by two experts of the law-firm that has collaborated jointly with other partners of the project. The gold standard is comprised of a set of 150 labor law questions in Spanish language related to the Workers Statute. Each question has been related with the title of the article in which the question is answered as shown in Table I.

The dataset with the rest of the documents used in this paper is published online[1] having followed the FAIR principles (findable, accessible, interoperable, reusable).

IV. Experiment

The experiment has been performed over a particular use case of Information Retrieval in Spanish language in the context of the European Project Lynx. The use case has been reported by a small business model of Cuatrecasas which is oriented to use the Spanish Workers Statute to solve client queries. The experiment is comprised of three main components. The first one is the document representation, how the corpora is represented and its format. Secondly, the database and its configuration properties. Finally, the description of the terminology extraction process.

[1]https://zenodo.org/record/4256718#.X6bIbGhKiUk

A. Document Representation

In the context of the European Project, the data model used for documents is the Legal Knowledge Graph (LKG) Ontology[2]. Lynx Documents are compliant with NIF (NLP Interchange Format)[3] specification and heavily reuses ELI[4] metadata elements. Lynx Documents may be grouped in Collections, and may be enriched with Annotations. Lynx Documents are the basic information units (pieces of text) and are described with Metadata and they are structured in Lynx Document Parts. Moreover, Collections groups documents with any logical relation but it can be achieved with metadata fields (`partOf` to reflect that a Lynx Document is part of another Lynx Document). An example Lynx Document is presented below.

In the experiment, the Spanish Workers Statute has been transformed into 130 Lynx Documents, one per article represented in the Statute

```
{
    "@context": "http://lynx-project.eu/doc/jsonld/lynxdocument.
        json",
    "id": "BOE-A-2015-11430_Part_69",
    "type": [ "lkg:Legislation",  "nif:Context", "lkg:
        LynxDocument" ],
    "text": "Articulo 67. Promocion de elecciones y mandato
        electoral.......",
    "metadata": {
      "skos:prefLabel": "Articulo 67 en Real Decreto Legislativo
          2/2015, de 23....",
      "lkg:partId": "BOE-A-2015-11430#offset_225601_231131",
      "jurisdiction": "ES",
      "language": "es",
      "title": {
        "es": "Articulo 67. Promocion de elecciones y mandato
            electoral."
      },
      "lkg:partOf": "BOE-A-2015-11430",
    },
    "parts": [],
    "annotations": [],
    "offset_ini": 0,
    "offset_end": 5530,
    "translations": {
      "en": "Article 67. Election promotion ..."
    }
  },
```

B. Database

The experiment is oriented to work with the document-oriented database Elasticsearch[5]. Elasticsearch organized documents in Indexes and these documents and their fields, which are represented in JSON format, can be configured to specify the how the values of the fields are stored and indexed in the database. For instance, for indexing text fields, usually the process involves the use of a tokenizer over the text. Moreover, a stemming process over resulting tokens is performed

[2]https://lynx-project.eu/doc/lkg/
[3]https://github.com/NLP2RDF/ontologies
[4]https://op.europa.eu/en/web/eu-vocabularies/eli
[5]https://www.elastic.co/

	Question	Document	Paragraph
Spanish	¿Qué situaciones interrumpen el periodo de prueba?	Artículo 14. Período de prueba	Las situaciones de incapacidad temporal, riesgo durante el embarazo, maternidad, adopción, guarda con fines de adopción, acogimiento, riesgo durante la lactancia y paternidad, que afecten al trabajador durante el periodo de prueba, interrumpen el cómputo del mismo siempre que se produzca acuerdo entre ambas partes.
English	What situations disrupt the trial period?	Article 14. Probationary Period.	Situations of temporary incapacity, maternity and the adoption or fostering of children affecting the worker during the probationary period interrupt the computation of the term, provided that agreement is reached between both parties.

TABLE I
Sample of the information captured in the Spanish Worker Statute Dataset: A question, the document where it is answered and the paragraph which contains it. In Spanish and English languages.

to ease the matching over variable tokens (e.g., cats, catlike, and catty have a common stem which is cat). Elasticsearch has stemmers for different languages such as English and Spanish.

Lynx Documents in their JSON format can be directly consumed by Elasticsearch. Also, for improving the results of the indexing process, a specific indexing configuration has been created for each language used in the context of the project. Tokenizers and stemmers for English and Spanish languages have been prepared for the documents of the Workers Statute.

Elasticsearch will use the fields of 'metadata.title' and 'text' for matching and scoring the queries.

C. Terminology Generation

As part of the Lynx project tasks, a terminology of the most relevant terms in the Spanish Workers Statute has been created. A key process within Information Retrieval is terminology extraction, that allows the identification of the most relevant terms within a document. For extraction process, we tested two different systems: the proprietary service provided by Tilde[6] as Lynx partner, that combines linguistic and statistical extraction; and an statistical open source application, TBXTools [6], which can be easily implemented and modified, when necessary. As a result, we obtained a combination of statistically and linguistically extracted terms that, with little post-processing, served as the basis for the generation of a terminology of labour law.

The list of retrieved terms was afterwards enhanced with translations in English, German and Dutch, synonyms, definitions and relations retrieved from the following existing language resources, being some of them close and proprietary, and others published in Semantic Web formats as part of the Linguistic Linked Open Data cloud[7]:

- Unesco Thesaurus: a controlled thesaurus containing mostly terms on social matters, structured in RDF and available to browse and download[8].
- EuroVoc: this was an originally hand-crafted thesaurus that was also transformed into RDF [1] and it is now available through an SPARQL endpoint[9] maintained by the Publications Office of the European Union.
- InterActive Terminology for Europe (IATE): this is a public resource containing terminology gathered by professionals in the European Union. The most recent version is available through a JSON API[10], although a previous version was converted into RDF following the lemon model [2].
- Lexicala: this API[11] is provided by KDictionaries, also part of Lynx consortium, and it contains general multilingual dictionaries.
- Wikidata: it is an open and collaborative knowledge base that is available through a user interface and also through a SPARQL query service[12].

Finally, the enhanced terminology was subsequently represented in SKOS[13], manually curated by domain experts from Cuatrecasas and published with open license in Zenodo. The resulting terminology is publicly available at[14]

V. Evaluation

The evaluation has been addressed by three experiments in relation of how the query is sent to the Elasticsearch database. The evaluation has been focused only the Spanish language because the experimentation is aligned with Cuatrecasas contribution and results in the project. The first experiment consisted in sending the question as it is written in the gold standard. In

[6]https://term.tilde.com/
[7]https://linguistic-lod.org/llod-cloud
[8]http://vocabularies.unesco.org/browser/thesaurus/en/
[9]http://publications.europa.eu/webapi/rdf/sparql
[10]https://iate.europa.eu/developers
[11]https://api.lexicala.com/
[12]https://query.wikidata.org/
[13]https://www.w3.org/TR/swbp-skos-core-spec/
[14]https://zenodo.org/record/3843561

	Perfect	Correct	Others	Not Found
Experiment 1. Q	66% (99 queries)	18% (27 queries)	11% (17 queries)	4% (6 queries)
Experiment 2. Q+ET	66% (99 queries)	18% (28 queries)	10% (15 queries)	4% (7 queries)
Experiment 3. Q+ET+ATE	68% (103 queries)	16% (24 queries)	10% (15 queries)	4% (7 queries)

TABLE II
Evaluation results for each experiment. Results are presented as a percentage from 0 to 100 and the number in the parenthesis represents the number of queries. Experiment 1 only takes into account the original query. In Experiment 2 the query is enriched with the Extracted Terminology and Experiment 3 includes the Automatic Term Extraction with Rake.

the second one, the question is enriched repeating those terms that are in the query and in the extracted terminology (relevant terms of the domain). Finally, in the last experiment the query is enriched as in the previous experiment and also repeating the query terms that have been identified by the library Rake[15]. The target of the library is to identify the important terms (nominal chunks) that appear in the query to reflect their importance in the scoring process.

The purpose of repeating terms in the query is to trunk the scoring values of the Elasticsearch search engine (coordination factor), increasing the value of these terms when the natural language query is sent.

The results for each query is a list of up to ten documents which are classified into four groups: perfect, correct, others and not found. Perfect means that the target document appears in the first result of the list (the most important one), correct means that it appears in the first four results, others means that the target document appears in the rest of the list and if it does not appear the list is classified as not found. The experiment results are presented in Table II.

The evaluation shows that even with the simple query in Experiment 1 the results reach a high performance in perfect and correct classifications (66% and 18%). The Elasticsearch stemming process for Spanish language has been critical to identify the correct documents in the database because the words of the query usually do not match perfectly with the words reflected in the documents due to the derived forms that are present in Spanish language (e.g., *trabajador* masculine worker, *trabajadora* feminine worker, *trabajadores* masculine workers, *trabajadoras* feminine workers).

Surprisingly, the use of the extracted terminology in Experiment 2 has not affected the overall results; only one question has passed from the category others to correct. Future experiments will analyze the use of the synonyms reflected in the terminology to enrich the query. However, the combination of the extracted terminology and the automatic term extraction over the query has reached the highest performance score with 103 documents (68%) classified as perfect.

[15]https://pypi.org/project/rake-nltk/

VI. Conclusions and Future Work

This work presents a new bilingual dataset oriented for the natural language processing tasks of Information Retrieval and Question Answering. The dataset is comprised of 150 labor law queries of the Spanish Workers Statute and it has been created in collaboration with domain experts of the law firm Cuatrecasas. Moreover, a simple Information Retrieval experiment has been developed in the context of a use case of the Lynx project in which the dataset is used for the evaluation. The dataset and the documents used in the experiment are distributed openly to the community.

As presented before, further experiments will exploit the use of synonyms extracted from the sources of information that are used to enrich the terminology. The main goal is to validate if these sources of information contain the correct synonyms that are used by end-users. Regarding the dataset, the main target is to increase the number of questions and answers to reach a trainable dataset that can be used by new state-of-the-art techniques such as BERT which requires a high amount of data.

VII. Acknowledgments

This paper has been funded by the project European project Lynx (Nº780602) and the European project Prêt-à-LLOD (Nº825182).

References

[1] M. L. Alvite Díez, B. Pérez León, M. M. Martínez González, D. J. Vicente Blanco, et al. Propuesta de representación del tesauro eurovoc en skos para su integración en sistemas de información jurídica. 2010.

[2] P. Cimiano, J. P. McCrae, V. Rodríguez-Doncel, T. Gornostay, A. Gómez-Pérez, B. Siemoneit, and A. Lagzdins. Linked terminologies: applying linked data principles to terminological resources. In *Proceedings of the eLex 2015 Conference*, pages 504–517, 2015.

[3] Y. Kano, M.-Y. Kim, R. Goebel, and K. Satoh. Overview of coliee 2017. In *COLIEE@ ICAIL*, pages 1–8, 2017.

[4] A. Mandal, K. Ghosh, A. Bhattacharya, A. Pal, and S. Ghosh. Overview of the fire 2017 irled track: Information retrieval from legal documents. In *FIRE*, 2017.

[5] C. D. Manning, H. Schütze, and P. Raghavan. *Introduction to information retrieval*. Cambridge university press, 2008.

[6] A. Oliver and M. Vàzquez. Tbxtools: a free, fast and flexible tool for automatic terminology extraction. In *Proceedings of the International Conference Recent Advances in Natural Language Processing*, pages 473–479, 2015.

[7] D. Samy, J. Arenas-García, and D. Pérez-Fernández. Legal-ES: A set of large scale resources for Spanish legal text processing. In *Proceedings of the 1st Workshop on Language Technologies for Government and Public Administration (LT4Gov)*, pages 32–36, Marseille, France, May 2020. European Language Resources Association.

[8] M. Saravanan, B. Ravindran, and S. Raman. Improving legal information retrieval using an ontological framework. *Artificial Intelligence and Law*, 17(2):101–124, 2009.

Towards Document Entity Recognition using Close Domain Transfer Learning

Adrián Inés, César Domínguez, Jónathan Heras, Eloy Mata and Vico Pascual
Department of Mathematics and Computer Science
Universidad de La Rioja
Logroño, Spain
{adrian.ines, cesar.dominguez, jonathan.heras, eloy.mata, mvico}@unirioja.es

***Abstract*—Nowadays, document data analysis is an increasing important task due to the large number of generated documents. Due to the increasing volume of documents, their manual analysis is a tedious and time-consuming process; and, therefore, automatic methods have arisen to deal with this task. Nowadays, a feasible approach for automatic document analysis is the application of deep learning methods. However, using these techniques requires a large number of annotated images, which can be difficult to obtain. This problem can be solved by using transfer learning, a technique that allows us to reuse the knowledge acquired from a different task. In this work, we have studied three different transfer learning approaches to create object detection models for the recognition of document entities such as text, titles, tables or figures. First, we have applied transfer learning from models pretrained in general detection tasks related to natural images. Second, we have studied an approach by training the backbones of the detection models in a close domain, using a document classification dataset. Finally, we have performed a close domain transfer learning training; that is, we have trained a model in a document detection dataset, and, then, we have retrained it in the target task. We have carried out a thorough analysis of each of the 3 approaches using 3 different detection architectures, and 5 document detection datasets. Our results show that, the close domain transfer learning approach can improve the performance of models between a 3% and a 20%.**

***Index Terms*—Object Detection, Transfer learning, Deep Learning, Document images.**

I. Introduction

Document analysis has become increasingly important due to the rapid growth in the number of available documents [4]. Every day thousands of documents, such as reports, forms, emails or invoices are either manually or automatically generated. Among the document analysis tasks, we can distinguish the classification of documents [3]; the detection of document entities, such as text, titles, tables or figures, within documents [11]; or the segmentation of document entities [5]. When the volume of documents increases, the manual analysis of documents becomes a very demanding task both in time and human resources. For that reason, analysing documents automatically is instrumental to exploit the information that is stored in them [6].

This work was partially supported by Ministerio de Economía y Competitividad [MTM2017-88804-P], and ADER [2017-I-IDD-00018], and Ministerio de Ciencia e Innovación and FEDER Funds [RTC-2017-6640-7], and FPU Grant 16/06903 of the Spanish MEC.

Currently, we can find different automatic techniques for document analysis. Among these techniques, we can highlight Optical Character Recognition (OCR) [22], computer vision techniques [12], and natural language processing methods [35]. In addition, deep learning has been successfully employed for document analysis [28]. The main problem with deep learning techniques is the large amount of data that is required to make them work properly [1]. At first sight, this might not seem a problem, due to the large number of documents that are generated in a daily basis; however, the manual annotation of these documents, a step that is instrumental to use supervised deep learning techniques, requires a considerable amount of time.

In the context of document entity recognition, the main approach employed to deal with the lack of enough annotated images consists in reusing a previously pre-trained model [16], [18], [19], [34]. Such models are either trained using an unsupervised regime from unlabelled images and taking into account visual, textual and layout features [16], [34]; or by employing models trained on a different domain that are adjusted to a particular task using either visual, textual or layout features [18], [19]. The advantage of the former models is that they are specifically designed for document understanding tasks; however, they require lots of resources to be trained (for instance, LayoutLM takes 170 hours per epoch using 8 NVIDIA Tesla V100 32GB GPUs) and are generally difficult to employ. On the contrary, the latter models are based on widely employed and standard deep learning architectures that are not as computationally demanding as the former models, but whose performance is worse. In this paper, we investigate how we can improve the latter models by selecting an adequate initial domain.

Models trained on a different domain that are adjusted to perform a different task belong to the category of transfer learning methods [25]. This technique makes possible to reduce the number of annotated images needed to train a deep learning model to solve a particular task by re-using the knowledge acquired in a different task. The main drawback of this technique appears when the tasks come from far domains. In most cases, transfer learning is applied from models that have been pretrained in large natural image datasets, like, ImageNet [7] for classification, and COCO [20] and PASCAL-VOC [8] for object detection; but, transfer learning from

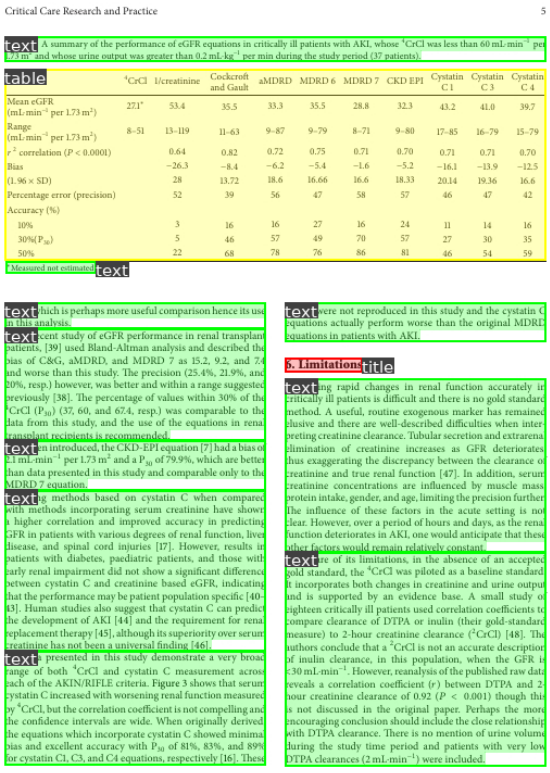


Fig. 1. Entity recognition on documents task example.

natural images to, for instance, medical images has several limitations due to the differences between these kinds of images [21], [24]. The hypothesis that we investigate in this work is that an adequate selection of a relevant source task can boost the performance of entity recognition on documents, see Figure 1.

The main contribution of this work is the study of different transfer learning approaches to create detection models for document entity recognition based on visual features. In particular, we have studied three different approaches that are presented in Section II. In Section III, we have explained the results obtained from the 3 approaches applied to 4 document entity recognition tasks. Finally, we end the paper drawing some conclusions from this study, and proposing new possible approaches in Section IV.

II. Materials and Methods

Object detection is a computer vision task that aims to locate the position of multiple objects in an image, and also provides the class of such objects. Currently, object detection tasks are mainly tackled using deep learning architectures that consist of a classification backbone and a head in charge of generating the detection boxes. Since training from-scratch these architectures is time-consuming and requires a considerable amount of resources, transfer learning is usually applied. Transfer learning for object detection can be performed either by loading a pretrained backbone, or by loading the whole model. In both cases, the model is later fine-tuned after the loading step. In this work, we have studied 3 different transfer learning approaches (traditional transfer learning, close backbone transfer, and, close architecture transfer) for document entity recognition compared with a from-scratch training process. For our experiments, we have used 3 deep learning architectures that are FasterRCNN [27], EfficientDet [32] and YOLO v4 [2]. FasterRCNN and EfficientDet are implemented in PyTorch [23] and have been trained thanks to the functionality of the Fastai [17] and IceVision[1] libraries using a GPU Nvidia RTX 2080 Ti. YOLO v4 is implemented in Darknet [26] and has been trained using the Darknet utilities and the GPU specified before.

These architectures have been tested on 5 different document entity detection datasets. In particular, we have used a large dataset and 4 small datasets to test the performance of the various transfer learning approaches. In Table I, we can see a description of the 5 datasets and the number of images that we have used for training and testing. We briefly describe these datasets as follows. The PubLayNet dataset, is a large dataset for document layout analysis that contains images of research papers and articles and annotations for various elements in a page (text, title, list, tables and figures). This dataset contains $335,703$ training images and $11,405$ test images of size 512×512. The ICDAR2017 dataset [10] consists of $2,000$ images of scientific papers with 3 kinds of objects to be detected: formulas, figures and tables. The FUNSD dataset [13] consists of 199 fully annotated forms with 4 kind of objects to be detected: headers, questions, answers and others. The UNLV dataset [29] contains 427 examples in scanned image format in which tables have to been detected. And, the Marmot dataset [9], that contains $2,000$ pages in PDF format where tables from research papers have to be detected.

Using these datasets, we have trained the aforementioned models by using three different transfer learning approaches and a scratch approach, that is, with random initial weights for both, the detection head and the backbone of the detection architectures, see Figure 2. The rest of this section is devoted to present the three different transfer learning approaches.

A. Traditional transfer learning

In the first transfer learning approach, we have studied the classical transfer learning approach using models pretrained on natural images. Currently, we can find a lot of pretrained detection models, most of them are pretrained in general detection tasks, using datasets such as COCO [20] and Pascal-VOC [8]. Also, we can find detection models with a backbone pretrained in a general classification task using the ImageNet dataset [7]. These datasets have hundreds of thousands of natural images with numerous examples in each image. Therefore, our first approach consists in reusing the knowledge acquired by detection models in these large datasets.

[1]https://airctic.com/

Dataset	♯ Train	♯ Test	Classes	Entities	Format	Colour
PubLayNet [36]	335,703	11,405	5	Texts, Titles, Lists, Tables and Figures	JPG	Yes
ICDAR [10]	1,200	400	3	Formulas, Figures and Tables	JPG	Yes
FUNSD [13]	149	50	4	Headers, Questions, Answers and Others	PNG	No
UNLV [29]	302	101	1	Tables	JPG	No
Marmot [9]	754	252	1	Tables	JPG	Yes

TABLE I
DESCRIPTION OF THE DATASETS EMPLOYED IN OUR EXPERIMENTS AND THE SPLIT MADE FOR TRAINING AND TESTING.

In particular, we have considered detection models with a backbone pretrained in the ImageNet challenge and a detection head with randomly initialised weights, see Figure 2. Then, we have fine-tuned these models using the new dataset. The Faster-RCNN and EfficientDet models have been trained using a two-stage transfer-learning method similar to the one presented in [17]. Specifically, in a first stage, we freezed the pretrained backbone of the model and trained the head of the detection model for two epochs. In the second stage, we unfreezed the whole model and retrained the model with the new data for 15 epochs using a suitable learning rate. In particular, we selected the learning rate that decreases the loss to the minimum possible value using the algorithm presented in [31]. Moreover, we employed early stopping based on monitoring the valid loss, and data augmentation [30] (using flips, rotations, zooms and lighting transformations) to prevent overfitting. For the YOLO v4 model, we have trained it by loading the pretrained ImageNet backbone, and then fine-tuned the whole model by one step training process with $12,000$ steps. We used 1e-3 as learning rate, and also applied early stopping and data augmentation to avoid overfitting.

B. Close backbone transfer

As we have previously explained, detection models consist of a classification backbone, and a head that allows us to generate the detection boxes. The idea of our second transfer learning approach is to perform a two-stage training process. First, we have trained a classification model in a document classification task. And, in a second stage, we have employed the methods presented in the previous section, but using the specific backbone trained on the previous stage, see Figure 2.

For the first stage of the process, we have trained three different families of classification architectures that are used as backbones by the detection models: ResNet50 [15] (for the Faster-RCNN architecture), EfficientNet-b2 [33] (for the EfficientDet architecture) and CSPDarknet53 [2] (for the YOLO architecture). These models have been trained using a two-stage transfer learning procedure similar to the procedure explained in the previous section. In the first stage, we replaced the last layers of the model (that is, the layers that give us the classification of the images), with a new one adapted to the number of classes of each particular dataset. Then, we trained these new layers (the rest of the layers stayed frozen) for two epochs. In the second stage, we unfreezed the whole model and retrained all the layers of the model with the new data for 50 epochs using a suitable learning rate [31].

The dataset selected to train these models was the RVL-CDIP dataset [14], a document classification dataset with 16 classes that consists of $400,000$ grayscale images of size 256×331 with $25,000$ images per class. These images are split into $320,000$ training images, $40,000$ validation images, and $40,000$ test images. In our experiments we have used the $320,000$ training images to train the models and the $40,000$ validation images as testing set. Also, we have resized the images to size 512×512. The trained models obtained the following accuracy: ResNet50, 0.9116; EfficientNet-B2, 0.9227; and CSPDarknet53, 0.8946.

C. Close architecture transfer

The last approach consists in training a detection model pretrained on a document entity detection task. To this aim, we have trained a detection model in a document entity detection task where enough annotated images where available; and, subsequently, we have retrained such model with a smaller target dataset, see Figure 1. The goal is to learn the general characteristics of detecting entities in documents using a sufficiently large dataset; and, later, refine the acquired knowledge for particular tasks in this domain.

In order to train the first detection model of this approach, we have applied the two methods presented in the previous sections to the PubLayNet dataset, and also trained the models from-scratch. The results obtained for this dataset can be found in Table II, where we can notice that, for all the models, the traditional transfer learning approach achieves better results than the other two training methods. Hence, those models have been employed for the second step of this transfer learning approach.

III. RESULTS AND DISCUSSION

In this section, we present the results for our experiments that are summarised in Table III. From those results, we can draw some general conclusions about the architectures and different training approaches.

A.1. Best architecture: From the conducted experiments, we can conclude that the YOLO architecture consistently produces the best results independently of the training approach. In particular, the YOLO model improve the performance of the FasterRCNN models between a 10% and a 20%, except for the Marmot dataset; and, between a 5% and a 160% for EfficientDet.

A.2. From scratch: Due to the limited number of training examples in each dataset, this approach generally offers the lowest results, the exception is for the YOLO models. In

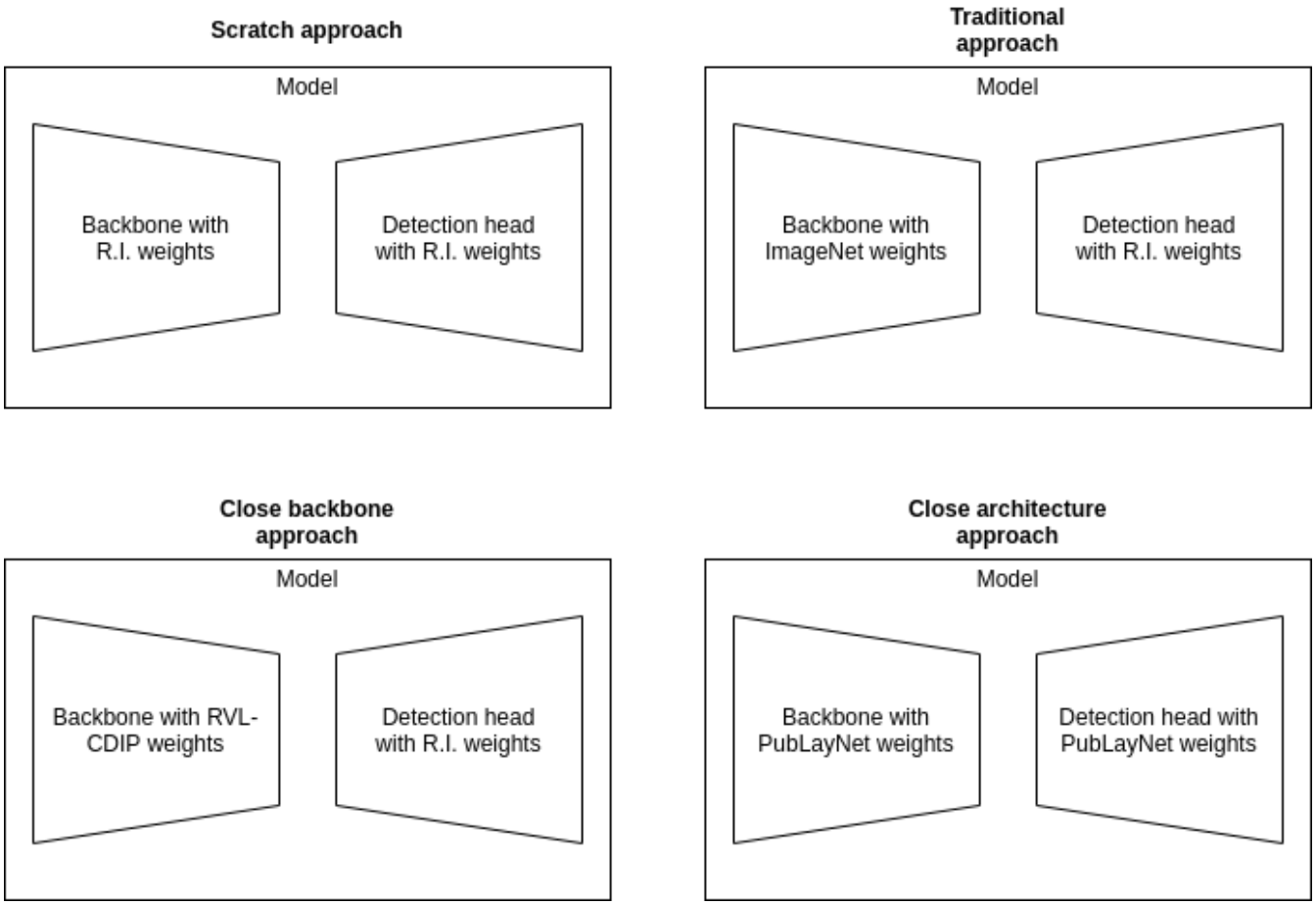


Fig. 2. Different training approaches employed in this work. R.I. stands for randomly initialised.

	From-scratch				Traditional				Close backbone			
Architecture	mAP	Prec	Rec	F1	mAP	Prec	Rec	F1	mAP	Prec	Rec	F1
FasterRCNN	0.90	0.97	0.98	0.98	**0.90**	0.97	0.99	0.98	0.90	0.97	0.98	0.97
EfficientDet	0.69	0.89	0.90	0.89	0.87	0.97	0.99	0.98	0.87	0.97	0.99	0.98
YOLO v4	0.90	0.98	0.96	0.97	0.90	0.98	0.96	0.97	0.88	0.97	0.95	0.96

TABLE II
MAP@[0.50:0.95:0.05], AND PRECISION, RECALL, AND F1-SCORE WITH IOU 0.5 ACHIEVED BY THE 3 STUDIED ARCHITECTURES BY TRAINING THEM FROM-SCRATCH, AND USING THE TRADITIONAL TRANSFER LEARNING APPROACH AND THE CLOSE BACKBONE TRANSFER APPROACH IN THE PUBLAYNET DATASET. BEST MODEL IS IN BOLD FACE.

addition, the training time and the number of epochs used in this approach is much higher than in the rest of approaches. This is due to the fact the rest of the approximations begin with a certain knowledge that makes them converge faster.

A.3. Traditional training: As it is known from the literature, the traditional approach using models pretrained on natural images improves the results obtained from the models initialised with random weights. This happens because the models take advantage of the learned characteristics in the largest classification datasets. This improvement varies from a 13% to a 90%.

A.4. Close backbone pretraining: The results obtained applying this approach depend on the architecture employed. Namely, all the FasterRCNN models trained using this approach obtain worse results than those obtained when training the models from scratch; on the contrary, the EfficientDet and YOLO models obtain better or similar results to those obtained with the traditional training process.

A.5. Close architecture fine-tuning: We can conclude that this approach produces the best results for all datasets and models. This show us, that the characteristics learned in the close domain are useful to obtain good results in the target task. We can also notice from the results of Table III that the datasets where this approach is more beneficial are those whose entities are a subset of those found in the PubLayNet dataset.

		Scratch				Traditional				Close backbone				Close architecture			
	Architecture	mAP	Prec	Rec	F1	mAP	Prec	Rec	F1	mAP	Prec	Rec	F1	mAP	Prec	Rec	F1
ICDAR	FasterRCNN	0.51	0.79	0.87	0.83	0.67	0.86	0.90	0.88	0.46	0.77	0.89	0.82	0.72	0.90	0.93	0.91
	EfficientDet	0.45	0.65	0.88	0.75	0.54	0.73	0.94	0.82	0.54	0.73	0.94	0.82	0.63	0.81	0.96	0.88
	YOLO v4	0.79	0.93	0.91	0.92	0.77	0.92	0.87	0.90	0.78	0.92	0.90	0.91	**0.79**	0.93	0.91	0.92
FUNSD	FasterRCNN	0.11	0.26	0.45	0.33	0.22	0.42	0.60	0.50	0.05	0.14	0.32	0.20	0.25	0.46	0.65	0.54
	EfficientDet	0.06	0.14	0.36	0.20	0.09	0.21	0.44	0.28	0.09	0.22	0.44	0.30	0.12	0.26	0.49	0.34
	YOLO v4	0.29	0.66	0.67	0.67	0.27	0.68	0.69	0.69	0.32	0.71	0.72	0.72	**0.32**	0.68	0.7	0.69
UNLV	FasterRCNN	0.45	0.76	0.89	0.82	0.71	0.89	0.92	0.91	0.09	0.30	0.67	0.42	0.73	0.92	0.98	0.95
	EfficientDet	0.56	0.77	0.99	0.86	0.66	0.90	1	0.95	0.65	0.89	1	0.94	0.71	0.91	0.99	0.95
	YOLO v4	0.79	0.96	0.92	0.94	0.81	0.95	0.93	0.94	0.78	0.95	0.92	0.94	**0.81**	0.95	0.95	0.95
MARM.	FasterRCNN	0.02	0.07	0.52	0.12	0.76	0.96	1	0.98	0.02	0.07	0.34	0.12	**0.87**	0.96	0.98	0.97
	EfficientDet	0.64	0.82	0.97	0.89	0.77	0.91	0.99	0.95	0.76	0.91	1	0.95	0.80	0.91	1	0.95
	YOLO v4	0.83	0.95	0.93	0.94	0.83	0.94	0.96	0.95	0.82	0.94	0.93	0.94	0.84	0.94	0.94	0.94

TABLE III
MAP@[0.50 : 0.95 : 0.05] AND PRECISION, RECALL AND F1-SCORE WITH IOU 0.5 ACHIEVED BY THE FOUR ARCHITECTURES STUDIED USING THE THREE APPROACHES IN THE FOUR DIFFERENT DATASETS. BEST MODELS ARE IN BOLD FACE.

IV. CONCLUSIONS AND FURTHER WORK

In this work, we have studied different transfer learning approaches to build detection models for entity recognition on document images. In particular, we have examined 3 different transfer learning approaches with 3 different deep object detection architectures. The best results have been achieved with the close architecture transfer approach using the YOLO v4 architecture, except for the Marmot dataset, where the FasterRCNN model trained with the close architecture transfer approach achieved the best results. With this study, we can conclude that starting from a model trained in a large dataset, from a close domain, improves the performance of those models. In contrast, training a backbone of a detection model in a classification task, even in a close domain, does not have to improve the results. What is more, this can worse them. As further work, we plan to test other architectures and use other training techniques such as semi-supervised and self-supervised learning that can reduce the amount of annotated images required for training the deep learning models.

REFERENCES

[1] Asperti, A., Mastronardo, C.: The effectiveness of data augmentation for detection of gastrointestinal diseases from endoscopical images. In: 11th International Joint Conference on Biomedical Engineering Systems and Technologies. BIOSTEC 2019, vol. 2, pp. 199–205. SciTePress (2018)

[2] Bochkovskiy, A., Wang, C.Y., Liao, H.Y.M.: Yolov4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934 (2020)

[3] Borko, H., Bernick, M.: Automatic document classification. Journal of the ACM (JACM) **10**(2), 151–162 (1963)

[4] Bowen, G.A.: Document analysis as a qualitative research method. Qualitative research journal (2009)

[5] Brants, T., Chen, F., Tsochantaridis, I.: Topic-based document segmentation with probabilistic latent semantic analysis. In: Proceedings of the eleventh international conference on Information and knowledge management. pp. 211–218 (2002)

[6] Coffey, A.: Analysing documents. The SAGE handbook of qualitative data analysis pp. 367–380 (2014)

[7] Deng, J., et al.: ImageNet: A Large-Scale Hierarchical Image Database. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR09) (2009)

[8] Everingham, M., Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. Int. J. Comput. Vision **88**(2), 303–338 (Jun 2010). https://doi.org/10.1007/s11263-009-0275-4, https://doi.org/10.1007/s11263-009-0275-4

[9] Fang, J., Tao, X., Tang, Z., Qiu, R., Liu, Y.: Dataset, ground-truth and performance metrics for table detection evaluation. In: 2012 10th IAPR International Workshop on Document Analysis Systems. pp. 445–449 (2012). https://doi.org/10.1109/DAS.2012.29

[10] Gao, L., Yi, X., Jiang, Z., Hao, L., Tang, Z.: Icdar2017 competition on page object detection. In: 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR). vol. 01, pp. 1417–1422 (2017). https://doi.org/10.1109/ICDAR.2017.231

[11] Gatos, B., Danatsas, D., Pratikakis, I., Perantonis, S.J.: Automatic table detection in document images. In: International Conference on Pattern Recognition and Image Analysis. pp. 609–618. Springer (2005)

[12] Gatos, B., Danatsas, D., Pratikakis, I., Perantonis, S.J.: Automatic table detection in document images. In: International Conference on Pattern Recognition and Image Analysis. pp. 609–618. Springer (2005)

[13] Guillaume Jaume, Hazim Kemal Ekenel, J.P.T.: Funsd: A dataset for form understanding in noisy scanned documents. In: Accepted to ICDAR-OST (2019)

[14] Harley, A.W., Ufkes, A., Derpanis, K.G.: Evaluation of deep convolutional nets for document image classification and retrieval. In: International Conference on Document Analysis and Recognition (ICDAR) (2015)

[15] He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
[16] Hong, T., et al.: A pre-trained language model for understanding texts in document (2021), https://openreview.net/forum?id=punMXQEsPr0
[17] Howard, J., Gugger, S.: Fastai: A Layered API for Deep Learning. Information **11**, 108 (02 2020)
[18] Huang, Y., et al.: A YOLO-based Table Detection Method. In: 15th International Conference on Document Analysis and Recognition. ICDAR'19 (2019)
[19] Kerwat, M., George, R., Shujaee, K.: Detecting Knowledge Artifacts in Scientific Document Images - Comparing Deep Learning Architectures. In: 5th International Conference on Social Networks Analysis, Management and Security. pp. 147–152. SNAMS'18, IEEE (2018)
[20] Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) Computer Vision – ECCV 2014. pp. 740–755. Springer International Publishing, Cham (2014)
[21] Mensink, T., Uijlings, J., Kuznetsova, A., Gygli, M., Ferrari, V.: Factors of influence for transfer learning across diverse appearance domains and task types (2021)
[22] Mori, S., Suen, C.Y., Yamamoto, K.: Historical review of ocr research and development. Proceedings of the IEEE **80**(7), 1029–1058 (1992)
[23] Paszke, A., et al.: Pytorch: An imperative style, high-performance deep learning library. In: Advances in Neural Information Processing Systems 32, pp. 8024–8035. Curran Associates, Inc. (2019)
[24] Raghu, M., Zhang, C., Kleinberg, J., Bengio, S.: Transfusion: Understanding transfer learning for medical imaging. arXiv preprint arXiv:1902.07208 (2019)
[25] Razavian, A.S., Azizpour, H., Sullivan, J., et al.: CNN features off-the-shelf: An astounding baseline for recognition. In: IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 512–519. CVPRW'14, IEEE (2014)
[26] Redmon, J.: Darknet: Open source neural networks in c. http://pjreddie.com/darknet/ (2013–2016)
[27] Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: towards real-time object detection with region proposal networks. IEEE transactions on pattern analysis and machine intelligence **39**(6), 1137–1149 (2016)
[28] Schreiber, S., Agne, S., Wolf, I., Dengel, A., Ahmed, S.: Deepdesrt: Deep learning for detection and structure recognition of tables in document images. In: 2017 14th IAPR international conference on document analysis and recognition (ICDAR). vol. 1, pp. 1162–1167. IEEE (2017)
[29] Shahab, A., Shafait, F., Kieninger, T., Dengel, A.: An open approach towards the benchmarking of table structure recognition systems. In: ACM International Conference Proceeding Series. pp. 113–120 (06 2010). https://doi.org/10.1145/1815330.1815345
[30] Simard, P., Steinkraus, D., Platt, J.C.: Best practices for convolutional neural networks applied to visual document analysis. In: 12th International Conference on Document Analysis and Recognition. vol. 2, pp. 958–964 (2003)
[31] Smith, L.: Cyclical learning rates for training neural networks. In: IEEE Winter Conference on Applications of Computer Vision. pp. 464–472 (2017)
[32] Tan, M., Pang, R., Le, Q.V.: Efficientdet: Scalable and efficient object detection. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 10778–10787 (2020). https://doi.org/10.1109/CVPR42600.2020.01079
[33] Tan, M., Le, Q.: Efficientnet: Rethinking model scaling for convolutional neural networks. In: International Conference on Machine Learning. pp. 6105–6114. PMLR (2019)
[34] Xu, Y., Li, M., Cui, L., Huang, S., Wei, F., Zhou, M.: Layoutlm: Pre-training of text and layout for document image understanding. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD '20), pp. 1192-1200. ACM (2020). https://doi.org/10.1145/3394486.3403172
[35] Yi, J., Nasukawa, T., Bunescu, R., Niblack, W.: Sentiment analyzer: Extracting sentiments about a given topic using natural language processing techniques. In: Third IEEE international conference on data mining. pp. 427–434. IEEE (2003)
[36] Zhong, X., Tang, J., Yepes, A.J.: Publaynet: largest dataset ever for document layout analysis. arXiv preprint arXiv:1908.07836 (2019)

Runners detection based on trajectory analysis

Jonay Sarmiento-Ramírez
Escuela de Ingeniería Informática
Universidad de Las Palmas de Gran Canaria
Las Palmas de G.C., Spain
jonay.sarmiento102@alu.ulpgc.es

Javier Lorenzo-Navarro*
Instituto Universitario SIANI
Universidad de Las Palmas de Gran Canaria
Las Palmas de G.C., Spain
javier.lorenzo@ulpgc.es (0000-0002-2834-2067)

Modesto Castrillón-Santana
Instituto Universitario SIANI
Universidad de Las Palmas de Gran Canaria
Las Palmas de G.C., Spain
modesto.castrillon@ulpgc.es (0000-0002-8673-2725)

Elena Sánchez-Nielsen
Dpto. Ingeniería Informática y de Sistemas
Universidad de La Laguna
La Laguna, Spain
enielsen@ull.edu.es (0000-0003-2114-4137)

***Abstract*—In some outdoor sports events, spectators and participants are not located in clearly different ubications. One example of such events is any ultratrail competition where both the spectators and runners share the space, with just some limitations in aid stations, where the organization staff may be present. An automatic visual-based runner detection could be of great help in mass events to support existing timing systems. For this aim, a challenging issue is to differentiate between participants and the rest in any recorded locations because sometimes the race bib number, which is a distinguishing cue, is not visible. In this work, the trajectories of the people in the scenario are explored as an element to solve the problem. Firstly, the trajectories are extracted combining YOLOV4 person detector and DeepSort tracker, and later manually annotated as runner or spectator. Then, a set of features are extracted from the trajectories and different classification methods are tested. For the experiments, 26 videos of a real ultra-trail race were used and a total of 756 trajectories were extracted. The best result was obtained with a two-level stacked LTSM that yields an accuracy of 98%.**

***Index Terms*—Computer Vision, Trajectory Classification, Sports analysis.**

I. Introduction

Computer Vision techniques have been widely applied in the sports context, as evidenced by the publication of recent surveys on the topic [1], [2], and the regular organization of workshops related such as CVsports and MMSports. Given the huge budget present in some highly demanded sports, there are many applications focused on the achievement of statistics related to team players [2], [3], but also in providing ways to improve the individual athlete performance [2], [4]–[6]. In both cases, the aim is to increase team or individual performance. Other Computer Vision techniques, such as tracking, have evidenced a large reliability improvement in the last years, and therefore introduced as an aid for assisting ball sports referees [7], [8]. However, biometric techniques as those related to people identification or re-identification, have not been exploited to aid the organizers of long-term sports, such as ultra-running, in detecting fraudulent actions as a mismatch between the runner and the registered person, or if a race bib number (RBN) is carried out by different runners during the course. A characteristic of long-term sports, which sometimes last more than 12 hours, is the fact that participants can change their clothes and wear different accessories like glasses or cap during the course. In this sense, biometrics can aid in runner identification but, as mentioned above, it has hardly been applied in the described scenario.

Adopting a supervised scenario, annotated data are necessary. Typical captures from ultra-running competitions may contain participants, staff, spectators, etc. Assuming video acquisition, the first problem to solve is to distinguish between runners from the rest as this kind of event takes place outdoors where the spectators can be very close to the runners (Figure 1). Detection of RBN can be considered as a solution because only runners wear it, but as stated before, in ultra-running events participants might wear outfits that occlude the RBN or after many hours of race the RBN can be folded or deteriorated (Figure 2). In this paper, we explore an approach based on the different behavior shown by runners and spectators when video footage is available. Runners normally exhibit a particular motion pattern, keeping a constant velocity/speed and moving in a defined path, unlike spectators that are static or wandering.

The contributions of the work are: 1) the annotation of people tracklets as runner/non-runner captured during a real ultra-running competition, 2) the exploration of features extracted from their extraction, and 3) the exploration of different classification methods to discriminate between both classes.

II. Related works

As previously stated, tracking players allows the analysis of tactics in team sports like basketball, soccer, or football. In [9], Chen et al. present a method based on player tracking with a

This work has been partially funded by the Spanish Ministry of Economy and Competitiveness (MINECO) under project RTI2018-093337-B-I00, and the Gobierno de Canarias and the FEDER funds under project ProID2020010024. We would also like to thank Arista Eventos S.L.U. to allow us to record images during Transgrancanaria 2020 and grant us their usage for research purposes.
*Corresponding author



Fig. 1. Spectators and runners in an ultra-marathon trail race. ©Alexis Martín (Creative Common License CC BY-NC 2.0)



Fig. 2. Two situations where RBN recognition will fail: RBN partially occluded (top) and folded RBN (bottom)

Kalman filter to recognize three of the most frequently used screen strategies in basketball using a set of rules. Also for basketball matches, Chen et al. [10] make use of a graph-based tracking method that can deal with player occlusions though only trajectories are obtained and no further analysis is carried out. Even if the players are not tracked to get tactics but the ball 3-D position estimated from 2-D, Chen et al. [11] propose a method to classify nine set types in volleyball games. In [12], Gómez et al. compare two tracking methods for both volleyball players and other two methods for ball tracking concluding that integral histogram and trajectory growth exhibit the best performance respectively. Deep learning approaches have been also introduced in this task. In [13], Schwenkreis maps the positions on the court (x,y) of the handball players during 15 seconds in a matrix of dimension 30 x 900, adopting a Convolutional Neural Network (CNN) to classify into 80 tactical moves. In Stein et al. [14], a technique for analysis of soccer is proposed where three types of relevant analysis are realized: region-based, event-based and player-based. Apart from sports, trajectory analysis is of interest in other fields as predicting the behavior of pedestrians in streets [15] or for assisted and autonomous driving [16]. In [17], Turchini et al. describe an unsupervised technique based on Landmark Based Spectral Clustering (LSC) of previously extracted trajectories with improved trajectories (IDT). Turchini's work aims to find a similarity measure between human activities in videos instead of classifying those activities as we propose in this work.

The application in ultra-running is limited, basically focused on identifying participants [18], [19]. The identification problem has certainly been already considered in the marathon scenario, even if the conditions are more controlled, compared to ultra-running, the literature is wider. In most cases, the Racing Bib Number (RBN) recognition strategy has been adopted for identification or re-identification [20], with rare attempts using facial or body appearance [19], [21]. Starting from the work described in [22], which includes the most used RBN benchmark, other approaches have recently been presented proposing different solutions using handcrafted features and deep learning, sometimes after a previous face, upper body, or body detection [23]–[25], to restrict the region of interest, or directly searching for RBNs [26]–[28]. Unlike marathon competitions where the spectators are placed separated from the runners, normally on the sidewalk, in the ultra-running scenario, those restrictions are in general more relaxed, as there are not big limitations because the sporting event takes place mainly along trails and paths, see Figure 1.

III. Methodology

A. Dataset

The experiments described below were carried out using some footage of the TGC2020 dataset [19], which was gathered during a real ultra-running competition held in March 2020. Runners taking part in Transgrancanaria (TGC) Classic 128KM in its 2020 edition left from the start line at 11 pm on March 6, to reach the finish line before its closure 30 hours later.

Participants were captured in six different recording points (RPs) (RP0-RP5) along the race track. For such purpose, a Sony Alpha ILCE 6400 (16-50mm lens and configured at 1920x1080@50fps) was employed at the different spots in continuous operation for a limited time, see Table I.

In locations RP1, RP4, and RP5, the spectators are close to the runners' track and in some cases, they walk on the race path (Figure 3 left) or even run near her/him (Figure 3 right). Additionally in RP1, the illumination conditions are not ideal because it was night (Figure 3 right) and the detection of RBN does not exhibit a reliable performance [18].

In total, 26 fragments of video from RP1, RP4, and RP5 were used to extract the trajectories of the people to be analyzed. YOLOv4 [29] and DeepSORT [30] were adopted for people detection and for tracking the detected person frame by frame, respectively. The number of trajectories obtained after this stage was 756, of which 655 correspond to the spectators and 101 to runners. For each frame, labels and bounding boxes for each detected person are obtained. The number of frames contained in the trajectories is 224,033, corresponding to 180,810 to spectators and 43,223 to runners. In Figure 4 it

TABLE I
RECORDING POINTS LOCATIONS AND RECORDING TIMETABLE.

Location	Timing Point	Km	Start Rec. Time	Footage (frames)
RP1	yes	16.5	00:06	140,616
RP2	yes	27.9	01:08	432,624
RP3	no	84.2	07:50	667,872
RP4	yes	110.5	10:20	1,001,208
RP5	yes	124.5	11:20	1,462,056



Fig. 3. Top: Spectators walking on the race path (RP4). Bottom: Spectator running close to the runner to record him (RP1).



Fig. 4. Examples of labeled trajectories of runners (green) and spectators (red).

can be seen examples of runners (green) and spectators (red) trajectories for a clip recorded at RP4.

With a frame rate of 50 fps, those trajectories with less than 100 frames were ruled out because they correspond to trajectories that last less than 2 seconds which can be produced by artifacts or detection failures. After the previous preprocessing, 385 trajectories were obtained, corresponding to 301 to spectators and 84 to runners.

B. Trajectory features

As it was described in Section III-A, for each person the id and the bounding box in the frame are obtained with the DeepSORT method. From these inputs, the following handcrafted features are computed:

- Height: For each frame i-th, the height of the bounding box for each detected person id is obtained.
- Speed: For each frame i-th, the speed of each detected person id in the image is computed as the distance between the bounding box centroid of person id in the current frame, $centr_{bb}(id, fr_i)$, and the previous one divided by the inverse of the frame rate.

$$speed_{id}(i) = \frac{||centr_{bb}(id, fr_{i-1}), centr_{bb}(id, fr_i)||_2}{1/fps} \quad (1)$$

- Average speed: To avoid artifacts, the average speed of the five previous frames is also computed.
- Direction: For each frame i-th, the direction of the person id in the image is computed from the bounding box centroid of person id in the current frame, $centr_{bb}(id, fr_i)$, and the previous one.

C. Classical Approach

A first attempt to differentiate runners from spectators using trajectory features in the videos is performed with classical Machine Learning methods. These methods accept a descriptor of fixed length known as dimensionality. The number of frames of the trajectories differs depending on the duration, so it is necessary to condense the information of the whole trajectory into a fixed size descriptor. To this end, a Bag of Words approach is implemented and obtaining the codebook by discretizing the features described in Section III-B. Therefore, the speed and the height were discretized into 10 and 11 bins, respectively, using an equal-frequency strategy, and the direction feature was mapped into the eight main cardinal points (N, S, W, E, NW, NE, SW, SE). Making use of the computed codebook, each frame is mapped to a 29-dimension feature vector (codeword). Finally, the trajectory is represented by the histogram of the codewords of all the frames that made up the trajectory.

D. Deep Learning Approach

A trajectory by definition is a temporal signal that gives the position of the person along the time, so a method that takes into account this fact would be the logical approach. In this sense, Recurrent Neural Networks (RNNs) are designed to deal with temporal signals, therefore a Long Term Short Memory (LSTM) [31] is evaluated for classification. Thus, the dataset was converted into sequences that feed the LSTM, and those sequences must have the same length. The size of the sequence must be a tradeoff between being as long as possible to capture the different behaviors of runners and non-runners,

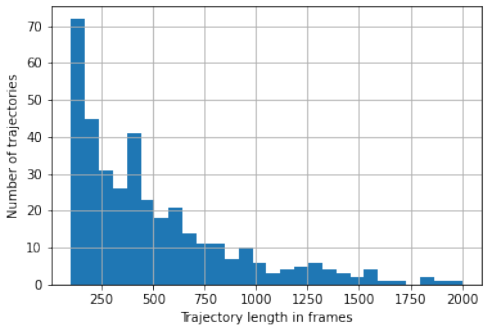


Fig. 5. Number of trajectories for different duration in frames

and not so long that the number of valid sequences in the dataset doesn't allow to train the network. Figure 5 shows the number of trajectories for different ranges of duration in frames, excluding those with less than 100 frames. A value of 400 frames was chosen so trajectories are restricted to have a length of 400 frames which implies sequences with a duration of 8 seconds. To compute these sequences, longer trajectories were divided into 400 frame sequences with a stride of 50 frames. Incomplete sequences were padded with zeros. In this case, the number of samples (sequences) is 3,218, of which 621 correspond to runners and 2,597 to spectators.

IV. Experiments and Results

The experimental setup for this problem was a holdout with 70% for training and 30% for testing. The classification methods used for the classical approach were: Decision Tree [32], Random Forest [33], Gradient Boosting [34], Naive Bayes [35] and Logistic Linear Regression. The hyperparameters for each method were: Gini criterion for the Decision Tree, 100 trees for the Random Forest, and logistic regression as loss function, and 100 stump trees as the base classifier for the Gradient Boosting.

As a reference, we perform a naive experiment to classify each frame as corresponding to spectator/runner. The frames are characterized with the features described in Section III-B. In this setup, the temporal component is not taken into account because each frame is considered independently of the rest. Therefore, in a trajectory, some frames may be classified as runner and the rest as spectator. It will serve as a baseline because no temporal information is utilized for classifying each person as runner/non-runner. In this setup, the training set is composed of 156,823 samples (126,584 spectators, 30,239 runners), and the test set is composed of 67,210 samples (54,226 spectators, 12,984 runners). Results are shown in Table II. It can be observed that even with no temporal information, an accuracy of 89% in classifying each frame as runner/spectator is achieved with a Random Forest classifier exhibiting a balanced performance in both classes according to precision and recall. The confusion matrix for the Random Forest is shown in Figure 6. It can be observed that the

TABLE II
Results of classical methods for frames (no trajectories).

Classifier	Accuracy	Precision	Recall
Decision Tree	0.84	0.83	0.84
Random Forest	**0.89**	**0.88**	**0.89**
Gradient Boosting	0.85	0.86	0.85
Naive Bayes	0.74	0.74	0.74
Logistic Regression	0.80	0.76	0.81

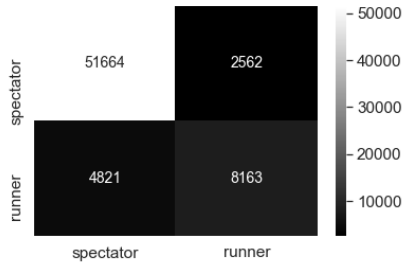


Fig. 6. Confusion matrix for Random Forest using frames.

TABLE III
Results of classical methods for trajectories using BoW approach.

Classifier	Accuracy	Precision	Recall
Decision Tree	0.75	0.80	0.75
Random Forest	**0.91**	**0.92**	**0.91**
Gradient Boosting	0.91	0.90	0.91
Naive Bayes	0.49	0.77	0.49
Logistic Regression	0.78	0.78	0.78

number of false positives for the runner class is almost the 40% of the runner frames as expected due that many frames that correspond to the runner trajectories are similar to the spectator frames. Though it has not been included in this work, the classification of the person can be obtained with a majority strategy, if the majority of the person's trajectory frames correspond to runner then is classified as a runner, otherwise is classified as a spectator.

When the trajectory information is condensed into a histogram following a BoW approach, two classifiers, Random Forest and Gradient Boosting, exhibit the highest performance with an accuracy of 91% (Table III), which implies an increase of a 2% over the baseline. Though both classifiers match in accuracy and recall, Random Forest improves a little in precision over Gradient Boosting. To note that in this case, the training and test set contains 269 samples (209 spectators, 60 runners) and 116 samples (92 spectators, 24 runners) respectively. Figure 7 shows the confusion matrix for the Random Forest classifier when the BoW approach is used for representing the trajectories. In this case, the percentage of false positives for the runner class is very similar to the previous setup which is due to the loss of temporal information because with the histogram the adjacency information among frames is lost.

Finally, to evaluate the use of the RNNs, the experimental

TABLE IV
RESULTS OF DIFFERENT LSTM TOPOLOGIES.

Topology	1st layer	2nd layer	FC layer	Accuracy	Precision	Recall
Single LSTM	32	-	32	0.92	0.87	0.68
Stacked LSTM	64	32	32	**0.98**	**0.95**	**0.93**
Stacked LSTM	64	64	64	0.96	0.88	0.87
Stacked LSTM	128	32	32	0.95	0.88	0.84

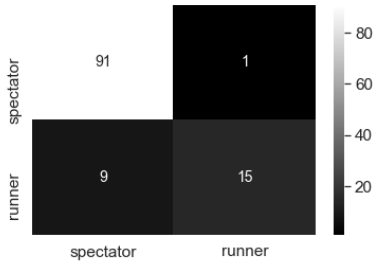


Fig. 7. Confusion matrix for Random Forest using BoW approach.

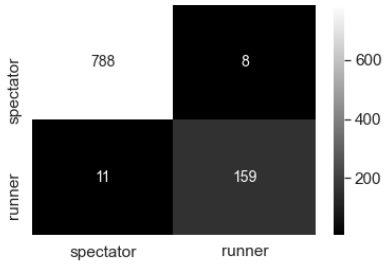


Fig. 8. Confusion matrix for Stacked LSTM (64-32 + 32)

setup was a holdout with 70% for training and 30% for testing, as with the classical approach. The training and test set has 2,252 samples (1,631 spectators, 621 runners) and 966 samples (796 spectators, 170 runners), respectively. The features considered as input to the neural network are the height, speed, average speed, and the eight main cardinal points. The three continuous features were normalized to the range $[0, 1]$. The chosen RNN architecture has been an LSTM and different topologies were considered. The upper layer of the LSTM feeds a fully connected layer with a sigmoid output function which is suitable for a bi-class problem as the one at hand. For all the configurations, the loss function was the binary cross-entropy and trained using the Adam optimizer with a learning rate of 0.001. To reduce the overfitting risk, a dropout of 0.5 was introduced in the training stage. As it is shown in Table IV, an accuracy of 98% was obtained with a two-level stacked LTSM with 64 cells for the first level, 32 for the second one, and 32 units in the fully connected layer. This result is a noticeable improvement over the best results obtained with classical approaches. As it can be observed in Figure 8, the confusion matrix for the best LSTM configuration shows a more balanced behavior in both classes as it was expected in comparison with the two previous experimental setups.

V. CONCLUSIONS

This work has presented the application of two different approaches to solving the problem of differentiating automatically runners from non-runners in ultra-marathon scenarios where the spectators may be close to participants, adding difficulties not present in other sports where participants and spectators occupy separated spaces. The two explored approaches are classical classifiers on the one hand, and RNNs on the other hand. In both cases, hand-crafted features obtained from the spectators and runners trajectories were used as input. As expected, the performance of RNNs, with an accuracy of 98%, was significantly superior to the classical approach because the temporal component is implicit in this kind of neural network. With the classical approach, only isolated frames can be considered or using the BoW approach to deal with variable length trajectories characterizing them with a fixed size descriptor. The experiments were carried out with real footage of a ultra-trail competition that covered both night and day scenarios.

REFERENCES

[1] T. B. Moeslund, G. Thomas, and A. Hilton, *Computer Vision in Sports*. Switzerland: Springer, 2014.
[2] G. Thomas, R. Gade, T. B. Moeslund, P. Carr, and A. Hilton, "Computer vision for sports: Current applications and research topics," *Computer Vision and Image Understanding*, vol. 159, pp. 3 – 18, 2017.
[3] H. Liu and B. Bhanu, "Pose-guided r-cnn for jersey number recognition in sports," in *Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. Long Beach, CA, USA: IEEE, 2019, pp. 2457–2466.
[4] G. Li and C. Zhang, "Automatic detection technology of sports athletes based on image recognition technology," *EURASIP Journal on Image and Video Processing*, vol. 2019, no. 1, p. 15, Jan 2019.
[5] C. Ning, "Design and research of motion video image analysis system in sports training," *Multimedia Tools and Applications*, March 2019.
[6] A. Rajšp and I. Fister, "A systematic literature review of intelligent data analysis methods for smart sport training," *Applied Sciences*, vol. 10, no. 9, 2020. [Online]. Available: https://www.mdpi.com/2076-3417/10/9/3013
[7] P. Kurowski, K. Szelag, W. Zaluski, and R. Sitnik, "Accurate ball tracking in volleyball actions to support referees," *Opto-Electronics Review*, vol. 26, no. 4, pp. 296–306, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1230340218301045
[8] M. Labayen, I. G. Olaizola, N. Aginako, and J. Florez, "Accurate ball trajectory tracking and 3d visualization for computer-assisted sports broadcast," *Multimedia Tools and Applications*, vol. 73, pp. 1819–1842, 2014.
[9] H.-T. Chen, C.-L. Chou, T.-S. Fu, S.-Y. Lee, and B.-S. P. Lin, "Recognizing tactic patterns in broadcast basketball video using player trajectory," *Journal of Visual Communication and Image Representation*, vol. 23, no. 6, pp. 932–947, 2012. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1047320312000995
[10] L.-H. Chen, C.-W. Su, and H.-A. Hsiao, "Player trajectory reconstruction for tactical analysis," *Multimedia Tools and Applications*, vol. 77, no. 23, p. 30475–30486, Dec. 2018.

[11] H.-T. Chen, W.-J. Tsai, S.-Y. Lee, and J.-Y. Yu, "Ball tracking and 3d trajectory approximation with applications to tactics analysis from single-camera volleyball sequences," *Multimedia Tools and Applications*, vol. 60, no. 3, p. 641–667, Oct. 2012.

[12] G. Gomez, P. H. López, D. Link, and B. Eskofier, "Tracking of ball and players in beach volleyball videos," *PLoS ONE*, vol. 9, 2014.

[13] F. Schwenkreis, "An approach to use deep learning to automatically recognize team tactics in team ball games," in *Proceedings of the 7th International Conference on Data Science, Technology and Applications (DATA 2018)*, Porto, Portugal, 2018, pp. 157–162.

[14] M. Stein, H. Janetzko, A. Lamprecht, T. Breitkreutz, P. Zimmermann, B. Goldlücke, T. Schreck, G. Andrienko, M. Grossniklaus, and D. A. Keim, "Bring it to the pitch: Combining video and movement data to enhance team sport analysis," *IEEE Transactions on Visualization and Computer Graphics*, vol. 24, no. 1, pp. 13–22, 2018.

[15] J. Liang, L. Jiang, J. Carlos Niebles, A. Hauptmann, and L. Fei-Fei, "Peeking into the future: Predicting future person activities and locations in videos," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019.

[16] A. Bhattacharyya, M. Fritz, and B. Schiele, "Long-term on-board prediction of people in traffic scenes under uncertainty," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[17] F. Turchini, L. Seidenari, and A. Del Bimbo, "Understanding and localizing activities from correspondences of clustered trajectories," *Computer Vision and Image Understanding*, vol. 159, pp. 128–142, 2017, computer Vision in Sports.

[18] P. Hernández-Carrascosa, A. Penate-Sanchez, J. F. O. Lorenzo-Navarro, and M. David; Castrillón-Santana, "TGCRBNW: A dataset for runner bib number detection (and recognition) in the wild," in *Proceedings International Conference on Pattern Recognition*, Milan, Italy, 2020.

[19] A. Penate-Sanchez, D. Freire-Obregón, A. Lorenzo-Melián, J. Lorenzo-Navarro, and M. Castrillón-Santana, "TGC20ReId: A dataset for sport event re-identification in the wild," *Pattern Recognition Letters*, vol. 138, pp. 355–361, 2020.

[20] P. X. Kamlesh, Y. Yang, and Y. Xu, "Person re-identification with end-to-end scene text recognition," in *Chinese Conference on Computer Vision*. Tianjin, China: Springer, 2017, pp. 363–374.

[21] A. Wrońska, K. Sarnacki, and K. Saeed, "Athlete number detection on the basis of their face images," in *Proceedings International Conference on Biometrics and Kansei Engineering*. Kyoto, Japan: IEEE, 2017, pp. 84–89.

[22] I. Ben-Ami, T. D. (Basha), and S. Avidan, "Racing bib number recognition," in *British Machine Vision Conference*. Surrey, UK: British Machine Vision Association, 2012, pp. 1–10.

[23] N. Boonsim, "Racing bib number localization on complex backgrounds," *WSEAS Transactions on Systems and Control*, vol. 13, pp. 226–231, 2018.

[24] S. Nag, R. Ramachandra, P. Shivakumara, U. Pal, T. Lu, and M. Kankanhall, "Crnn based jersey-bib number/text recognition in sports and marathon images," in *International Conference on Document Analysis and Recognition (ICDAR)*. Sydney, Australia: IEEE, 2019, pp. 1149–1156.

[25] P. Shivakumara, R. Raghavendra, L. Qin, K. B.Raja, T. Luc, and U. Pal, "A new multi-modal approach to bib number/text detection and recognition in marathon images," *Pattern Recognition*, vol. 61, pp. 479–491, 2017.

[26] A. Apap and D. Seychel, "Marathon bib number recognition using deep learning," in *11th International Symposium on Image and Signal Processing and Analysis (ISPA)*. Dubrovnik, Croatia: IEEE, 2019, pp. 21–26.

[27] W. M. de Jesús and D. L. Borges, "An improved stroke width transform to detect race bib numbers," in *Proceedings of the Mexican Conference on Pattern Recognition*. Puebla, Mexico: Springer, 2018, pp. 267–276.

[28] Y. C. Wong, L. J. Choi, R. S. S. Singh, H. Zhang, and A. R. Syafeeza, "Deep learning based racing bib number detection and recognition," *Jordanian Journal of Computers and Information Technology (JJCIT)*, vol. 5(3):, no. 3, pp. 181–194, 2019.

[29] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," *arXiv*, vol. 2004.10934, 2020. [Online]. Available: https://arxiv.org/abs/2004.10934

[30] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in *2017 IEEE International Conference on Image Processing (ICIP)*, 2017, pp. 3645–3649.

[31] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[32] J. R. Quinlan, "Induction of decision trees," *Machine Learning*, vol. 1, pp. 81–106, 1986.

[33] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, Oct 2001.

[34] J. H. Friedman, "Greedy function approximation: a gradient boosting machine," *Annals of statistics*, pp. 1189–1232, 2001.

[35] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification (2nd Edition)*. Wiley-Interscience, 2001.

Evaluación Estética de Fotografías vía Deep Learning y enfoques probabilísticos

Luis González Naharro*, José Miguel Puerta Callejón†, Fernando Rubio Perona‡ y Maria Julia Flores Gallego§
SIMD Lab I3A, UCLM
Albacete, España
Email: *Luis.GNaharro@uclm.es, †Jose.Puerta@uclm.es, ‡Fernando.Rubio@uclm.es, §Julia.Flores@uclm.es

***Abstract*—Una de las técnicas más populares en la actualidad para evaluar automáticamente la calidad estética de fotografías es emplear redes neuronales profundas entrenadas para tal propósito.**

Aunque esta técnica de evaluación resulta muy prometedora, existe un problema crucial en los conjuntos de datos utilizados para entrenar estas redes: no hay consenso a la hora de determinar cuándo una fotografía es, o no, de alta calidad estética. Los enfoques clásicos se basan en considerar una distribución de votos recibida en la fotografía por parte de usuarios, y tomar la media como indicador, pero ésta tiene el problema de que genera demasiada incertidumbre sobre las fotografías de calidad regular.

En este trabajo planteamos técnicas alternativas a la media para obtener las fotografías consideradas de alta calidad y las de baja calidad. Estos nuevos métodos de evaluación de la calidad de las fotografías a través de la distribución de los votos se comprobarán en dos modelos de red convolucional profunda basados en NIMA, cada uno de los cuales tendrá como salida, para una imagen dada, la estimación de su distribución de votos, y su clasificación estética binaria, respectivamente

***Index Terms*—Deep Learning, Visión Artificial, Clasificación Supervisada**

I. Introducción

La visión por ordenador es uno de los campos más importantes actualmente en el mundo de la Inteligencia Artificial. Su enorme cantidad de aplicaciones han hecho de este problema uno de los más atractivos para la comunidad científica, tanto por sus aplicaciones como por su dificultad.

De entre todos los problemas de visión artificial, en este artículo nos centramos en la evaluación automática de la calidad estética de fotografías. Esto es, poder determinar automáticamente si una fotografía es "bonita" o no. Este problema, aunque es altamente subjetivo, ha conseguido despertar un gran interés en la comunidad científica, puesto que tiene aplicaciones en las que las fotografías de calidad son cruciales, como en la publicidad, o en los buscadores de imágenes.

Dada esta subjetividad, lo más parecido a tener fotografías etiquetadas como de alta/baja calidad estética son las creencias de un conjunto de usuarios sobre la calidad estética de las mismas, usualmente en forma de votos (del 1 al 10, en forma de *likes*/*dislikes*, etc.). Una de las formas de resolver este tipo de problemas es tratar de transformarlos a datos supervisados de calidad, momento en el cual se podría utilizar cualquier método de aprendizaje supervisado.

Considerando el uso de redes convolucionales para llevar a cabo esta tarea (cuya arquitectura se puede ver en la Figura 1), es necesario disponer de conjuntos de imágenes lo suficientemente grandes como para que dichas redes se puedan entrenar correctamente. Uno de los conjuntos más importantes en la actualidad es AVA [1], destacable por la gran cantidad de imágenes que contiene.

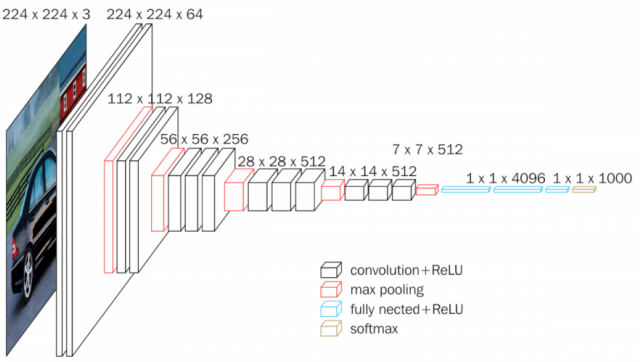


Fig. 1: Arquitectura básica de una CNN

Aun con conjuntos de datos así, sigue habiendo falta de consenso en cuanto a cómo determinar si una fotografía es de alta calidad estética o no. Casi todos los trabajos actuales [2]–[6] asumen que las fotografías con una puntuación media mayor que 5 son de alta calidad, mientras que las demás son de baja calidad. No obstante, como refleja la Figura 2, las puntuaciones medias de las imágenes en AVA están concentradas en torno al 5, lo cual es un problema para la clasificación de las fotografías de calidad media/regular, y hasta puede sesgar los resultados de las fotografías de alta calidad con predicciones cercanas al 5 [7].

Por lo tanto, el objetivo principal de este trabajo es analizar criterios alternativos a la media para determinar la calidad de una imagen. Nuestras propuestas se basan principalmente en aplicar *clustering* sobre las distribuciones de votos de AVA. Nos centraremos en el uso de modelos basados en **LDA**, en mixturas de **Gaussianas**, y en **K-medias**. Pretendemos alcanzar dos objetivos con estas técnicas: a) Dejar que el método en cuestión decida dónde debe estar el umbral de clasificación (más allá de usar la media como umbral), que además puede cambiar entre conjuntos de datos distintos; y b) ponderar mejor las calificaciones recibidas en los votos de una cierta imagen, puesto que las calificaciones centrales

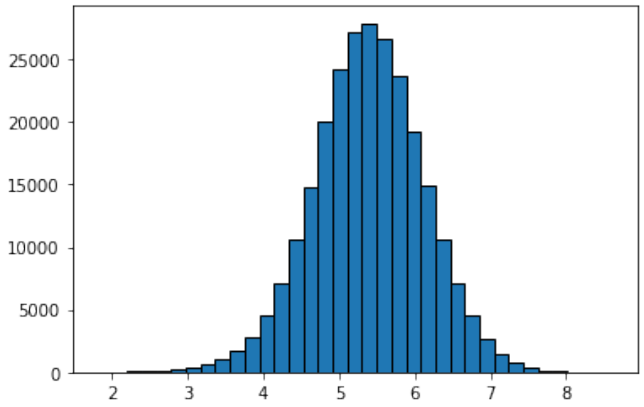


Fig. 2: Distribución de puntuaciones medias en AVA.

tienen demasiado peso y reflejan una alta incertidumbre sobre la imagen.

La estructura del resto del trabajo es la que sigue. En la Sección II se describirán el conjunto de datos y la arquitectura de CNN que se usará en este trabajo, además de discutir trabajos recientes sobre el mismo problema. En la Sección III se presentarán las técnicas de *clustering* que se van a estudiar. En la Sección IV se presentarán tanto los experimentos a realizar como los resultados obtenidos por cada uno de estos experimentos. Por último, en la Sección V se detallarán las conclusiones obtenidas de este trabajo, así como las tareas futuras a realizar.

II. Estado del arte

Por lo general, los conjuntos de imágenes más actuales proporcionan una distribución de votos de cada imagen. Aunque así se puede extraer información mucho más rica de las fotografías, esto también hace que surja el problema de cómo determinar si una fotografía es de alta calidad estética o no.

Originalmente, [8] y [9] reducen el problema a la construcción de características a partir de las imágenes, tales como histogramas de color o exposición. Después, utilizan estas características para hacer clasificación binaria, usando la media como punto de corte.

En [10], se propone utilizar características generales de las imágenes como *Scale-invariant feature transform* (SIFT) o GIST para evaluar la calidad estética, mejorando los resultados obtenidos hasta el momento.

A. Aesthetic Visual Analysis dataset (AVA)

AVA [1] es un conjunto de fotografías creado para resolver problemas de calidad estética en imágenes. Es de gran tamaño en comparación con otros como [8], [11] o [12]. Ha sido obtenido del sitio web de concursos fotográficos `www.dpchallenge.com`, y contiene aproximadamente 250000 imágenes de calidad variable. En este sitio web se llevan a cabo desafíos fotográficos diversos, donde las fotografías reciben votos del 1 al 10 por parte de usuarios *amateurs* y profesionales.

En AVA, cada fotografía tiene de media 210 votos, oscilando entre 78 y 549. Además, también incorpora información semántica sobre dichas imágenes, con 66 etiquetas textuales que describen información adicional de las imágenes. La Figura 2 muestra la distribución de los votos de AVA. En ella se puede observar el efecto que ya se ha descrito anteriormente: la alta concentración de puntuaciones medias en torno al 5.

B. Redes convolucionales para clasificación estética

Con la aparición de AVA, y gracias a su volumen de datos, se ha podido pasar de modelos tradicionales a modelos más potentes basados en CNNs. Una de las primeras propuestas es [13], basada en el entrenamiento desde cero de CNNs y con el enfoque de clasificación binaria, aunque debido al uso de la media como discriminador, sus mejoras no son tan significativas como en otros problemas de visión por ordenador.

En [14], los autores entrenan un modelo mediante una red neuronal siamesa mediante pares de imágenes. No obstante, este enfoque sigue basándose en la media, pues la ordenación entre imágenes se hace en base a la misma: por lo tanto, habrá mucha incertidumbre entre imágenes con medias cercanas al umbral de decisión.

En [5], los autores utilizan parches de imágenes para predecir su calidad estética, ajustando sus pesos durante el proceso de entrenamiento para mejorar el proceso de aprendizaje, dando más peso a los parches con errores de predicción. Sin embargo, de nuevo se utiliza la media como discriminador de imágenes de alta calidad estética y de baja calidad.

En [15], los autores proponen un método para poder predecir la puntuación estética de imágenes de tamaño variable. Los autores pasan las imágenes por una red convolucional, extrayendo *deep features* de los bloques convolucionales de dicha red, concatenándolas y utilizándolas como entrada de otra red neuronal que realiza la predicción de la calidad estética. Pero de nuevo se utiliza la media como discriminador.

C. Neural Image Assessment (NIMA)

NIMA [7] es una arquitectura de red neuronal desarrollada para el problema de clasificación estética de fotografías. Esta arquitectura intenta predecir directamente la distribución de votos de la imagen, utilizando *Earth Mover Distance* (EMD) como función de pérdida, la cual se utiliza en problemas en los que existe una ordenación entre las clases (en este caso, los votos de las imágenes). Esta función se define para una distribución de votos predicha $\hat{p}$ y su correspondiente *ground-truth* p como:

$$EMD(p, \hat{p}) = \sqrt{\left(\frac{1}{N} \sum_{k=1}^{N} \left(CDF_p(k) - CDF_{\hat{p}}(k) \right)^2 \right)}$$

donde N es el número de calificaciones de las distribuciones de votos, y $CDF(k)$ es la función de probabilidad cumulativa de una distribución de votos hasta la calificación k-ésima.

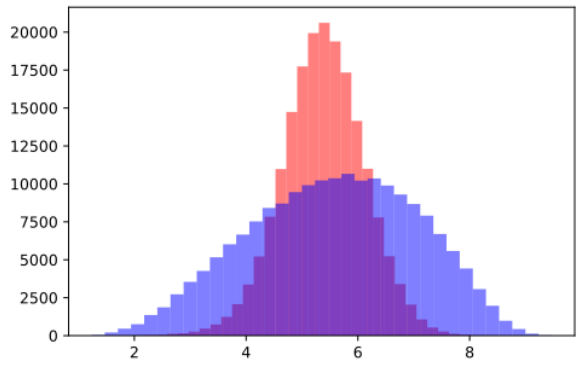


Fig. 3: Comparación de Distribuciones de AVA y su transformación a RF-IRF. Rojo: Original AVA. Azul: RF-IRF.

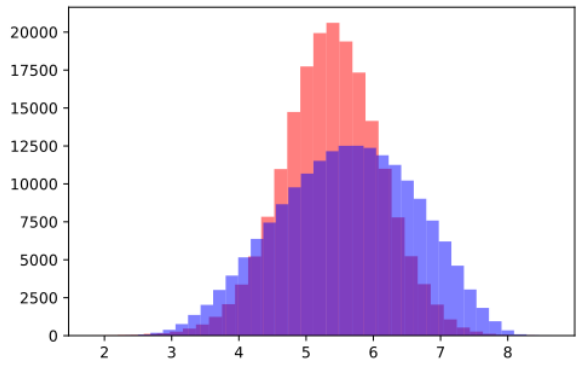


Fig. 4: Comparación de Distribuciones de AVA y su transformación a RF-IRF suavizado. Rojo: Original AVA. Azul: RF-IRF suavizado.

La arquitectura de NIMA utiliza una red neuronal para clasificación de imágenes como red base, en la que se cambia la última capa por una capa *fully-connected* de 10 neuronas, procesadas por una activación softmax, para transformar las predicciones de la distribución de votos a una distribución de probabilidad.

Esta arquitectura se entrena inicializando los pesos de la red principal mediante *transfer learning* sobre ImageNet. Tras este proceso, los pesos de la capa *fully-connected* se inicializan de forma aleatoria, y después se entrena la red entera para predecir correctamente la distribución de votos de las imágenes. La salida de la red es una distribución de probabilidad de cada una de las calificaciones de las imágenes. Así pues, se puede calcular la puntuación media de las predicciones hechas por la red, y hacer clasificación binaria para una imagen, considerando una puntuación media de 5 como punto de corte.

III. Modelando los Votos de los Usuarios de AVA

Es necesario desarrollar técnicas alternativas a la media que creen una separación mucho más clara y de calidad entre fotografías de alta y baja calidad estética, debido a la poca varianza que ésta presenta.

A. Modelos vectoriales de transformación de votos

En primer lugar, vamos a proponer dos modelos de transformación de los votos, basados en los modelos vectoriales TF-IDF [16] para representación de documentos. Estas transformaciones están diseñadas para disminuir la ponderación de los votos de mayor incertidumbre, y para aumentar la ponderación de los votos de menor incertidumbre.

1) Modelo Vectorial de Votos RF-IRF: El primero de estos modelos, que hemos denominado como **espacio vectorial Rating Frequency - Inverse Rating Frequency** (RF-IRF), se trata de una adaptación de TF-IDF para ponderar las calificaciones de la escala de los votos de cada imagen. Así, si una imagen tiene muchos votos en una calificación determinada [1], dicha calificación va a tener más poder discriminatorio contra otras imágenes que tengan una frecuencia baja en dicha calificación. No obstante, si dicha calificación es muy frecuente en todas las imágenes, se puede considerar que tiene un poder discriminatorio menor. Así pues definiremos:

- ***Rating Frequency* (RF)**: La frecuencia de una calificación dentro de una imagen. Una mayor frecuencia indicará una mayor ponderación de dicha calificación. Se calcula como $RF_f = R_{f,r}$, donde $R_{f,r}$ es la frecuencia de los votos de la imagen f-ésima de la colección de M imágenes para la calificación r
- ***Inverse Rating Frequency* (IRF)**: La frecuencia inversa total de una calificación en el conjunto completo de imágenes. Una alta frecuencia total de votos indicará un menor peso de la calificación correspondiente. Se calcula como $IRF_r = \left(\sum_{f=0}^{M} R_{f,r}\right)^{-1}$, donde r representa el *rating*/calificación, r.

Tras realizar esta transformación, se multiplican las componentes RF e IRF, se normaliza el vector resultante y se multiplica por el número original de votos, para obtener así la frecuencia esperada de cada calificación en este espacio alternativo de votos. El resultado de esta transformación se puede ver en la Figura 3. Nótese cómo esta transformación está consiguiendo aplanar la curva de la distribución original de votos.

2) RF-IRF suavizado: En la literatura es usual encontrarse con un enfoque suavizado de los modelos TF-IDF:

1) En primer lugar, se calcula el vector IRF como en el caso anterior, pero sin tomar su inversa:

$$RF_r = \left(\sum_{f=0}^{M} R_{f,r}\right)$$

[1]Mientras que las calificaciones son valores del 1 al 10 en los que puede haber votos, los votos son la cantidad concreta de calificaciones que tiene una imagen en cada calificación.

2) A continuación, se recalcula el conjunto de votos, tomando el logaritmo de cada elemento individual más uno, obteniendo así un conjunto de votos transformado:

$$lnRF_{f,r} = ln(1 + R_{f,r})$$

3) Se calcula el coeficiente N como la suma de todos los votos:

$$N = \sum_{f,r}^{M,N} R_{f,r}$$

4) Se recalcula el vector IRF_r como:

$$lnIRF_r = ln\left(\frac{N - RF_r}{RF_r}\right)$$

Por último, se calcula el conjunto de votos transformado de una manera similar a la anterior, se multiplican las componentes $lnRF \times lnIRF$, se normalizan y se multiplican por el número de votos de cada imagen para tener el número de votos esperado en este nuevo espacio. El resultado de esta transformación se puede ver en la Figura 4. Nótese que esta transformación está aplanando también la curva de votos, aunque de una manera más suave que RF-IRF.

B. Modelos de clustering

En segundo lugar, vamos a estudiar tres modelos de *clustering* de votos de los usuarios. El primero de estos modelos será el ya conocido algoritmo de K-medias [17], [18]. El segundo será un modelo de mixturas de Gaussianas [19]. Por último, describiremos otro modelo, basado en modelos generativos para descubrir tópicos en colecciones de documentos: *Latent Dirichlet Allocation* [20]. Nuestro objetivo es que estos modelos elijan el mejor punto de corte posible para las imágenes, sin tener que fijar un umbral a priori como la media.

Todos estos modelos de *clustering* generan como salida una distribución de probabilidad de dos componentes, donde cada componente indica el tipo de imagen (alta o baja calidad).

1) Modelo Generativo basado en Mixturas de Gaussianas: Una primera aproximación es utilizar una mixtura de Gaussianas de dos componentes, una para cada calidad de imagen, y suponer normalidad en la distribución de votos. La estimación de parámetros se puede hacer mediante *Expectation Maximization* (EM) [21].

2) Modelo Generativo Probabilístico de Votos basado en LDA: En segundo lugar, se utilizará un modelo generativo probabilístico de votos, diseñado de una manera muy similar al modelo de **Latent Dirichlet Allocation** (LDA) [20]. Originalmente, esta técnica se desarrolló para representar corpus de documentos, aunque se puede extrapolar a nuestro problema.

Este modelo gráfico factoriza la distribución conjunta de todas las variables del modelo como sigue:

$$\prod_{i=0}^{1} P(\beta_i|\nu) \prod_{f=1}^{M} P(\theta_f|\alpha) \left[\prod_{v=1}^{N} P(z_{f,v}|\theta_f) P(r_{f,v}|\beta; z_{f,v})\right]$$

A partir de todos los elementos, el modelo generativo muestrea el conjunto de votos para las imágenes como:

1) A partir de ν, para la calidad alta y la calidad baja, se genera una distribución de probabilidad de ratings o calificaciones. Esta distribución representa la probabilidad de cada calificación del 1 al 10, condicionada a que la fotografía sea de alta o baja calidad.
2) A partir de α se determina la probabilidad θ de que cada una de las fotografías sea alta o baja.
3) A partir de θ se determina si cada uno de los usuarios que han votado en una imagen han votado considerando que la fotografía es de alta calidad estética, o si han votado considerando que es de baja calidad. Esta asignación por voto y por fotografía se corresponde con Z.
4) Por último, a partir de la distribución de calificaciones/*ratings* dada la calidad (β) y a partir de la consideración/asignación de cada usuario sobre una fotografía, se generan las calificaciones R: la distribución de los votos de una imagen en función de su calificación.

Los parámetros de todas estas distribuciones se aprenden con *Online Variational Bayes Algorithm*, propuesto en el artículo original de LDA, e integrado en el *framework* de `scikit-learn` [22].

Una vez aprendido este modelo con los votos del conjunto de datos, el algoritmo ofrece dos salidas: por una parte, la distribución de cada fotografía en sus tópicos correspondientes (las probabilidades de que sea de alta o de baja calidad), y por otra parte, los pesos de cada calificación (de 1 a 10), condicionados a que la calidad sea alta o baja. No obstante, en este trabajo tan sólo nos centraremos en la primera salida.

IV. Experimentos y resultados

A. Experimentos

Para evaluar las propuestas realizadas en este artículo, vamos a realizar dos experimentos. En todos ellos se ha hecho una partición aleatoria de entrenamiento-validación-test de 0.736-0.184-0.08 de todo AVA, además de usar un *learning rate* de 10^{-5}, un tamaño de *mini-batch* de 128, y MobileNet como red base de NIMA.

B. Predicción de la distribución de votos

En primer lugar, vamos a evaluar el funcionamiento de las transformaciones RF-IRF y su variante suavizada, utilizando las distribuciones de votos resultantes como *ground-truth* sobre las que entrenar a NIMA para predecir **las distribuciones de votos**, y evaluando el rendimiento de estos modelos comparándolos tanto con dichas *ground-truth*, como con las predicciones de NIMA, como con la distribución de votos reales.

Los modelos que se van a utilizar en estos experimentos son tanto RF-IRF como su variante suavizada, además de un *baseline* para evaluar su rendimiento (el cual es totalmente equivalente a NIMA). De cada uno de estos modelos se han estimado las siguientes métricas:

- *Accuracy*.

TABLA I: Comparativas de los modelos de predicción de distribución contra conjunto de test

	Balanced accuracy	EMD media	Accuracy
BASELINE			
Votos reales	0.6948	0.0197	0.7749
RF-IRF			
Transformación	**0.6972**	0.0322	0.7397
Predicciones NIMA	**0.9103**	**0.0565**	0.8739
Votos reales	**0.7194**	**0.0631**	0.7676
RF-IRF SUAVIZADO			
Transformación	0.6488	**0.0207**	**0.7670**
Predicciones NIMA	0.8688	0.0644	**0.9341**
Votos reales	0.6635	0.0709	**0.7757**

- *Balanced accuracy*: Una variante de la *accuracy* que calcula el *accuracy* medio en todas las clases del problema.
- *EMD media*: mide la media de la distancia EMD, de todas las distribuciones predichas, entre la distribución predicha y aquella con la que se está haciendo la comparación.

Las dos primeras métricas se han obtenido calculando la media de las distribuciones predichas por los modelos y cortando por el valor 0.5 en imágenes de alta y baja calidad estética. Aunque se incluye la *accuracy*, esta división suele crear desbalanceos en la clase binaria, por lo que la *balanced accuracy* es una métrica más significativa.

Por cada conjunto de predicciones hecho por cada modelo, se han realizado las siguientes comparaciones:

- Comparación con la **nueva *ground-truth***: las predicciones se comparan con el conjunto de test, transformado de la misma manera que se ha transformado el conjunto de entrenamiento.
- Comparación con la **salida de NIMA**: las predicciones se comparan con la salida de NIMA para el conjunto de test, para comprobar que se está consiguiendo hacer algo similar a lo que hace NIMA.
- Comparación con los **votos reales**: las predicciones se comparan con los votos reales, para comprobar que las predicciones se parecen, en efecto, a la *ground-truth* original.

Los resultados de estas comparativas se pueden ver en la Tabla I[2]. En primer lugar, se puede observar que el *baseline* no es totalmente idéntico a NIMA. Esto posiblemente se deba a pequeñas diferencias dependientes de la manera de entrenar a la red.

Considerando todos los modelos, y atendiendo a la *balanced accuracy*, se puede observar que RF-IRF está consiguiendo una mejora importante en cuanto a los votos reales. También tiene buenos resultados en cuanto a la distancia EMD con respecto a los votos reales, aunque esta distancia es mayor que la del *baseline*. Por otra parte, RF-IRF suavizado está consiguiendo buenos resultados en *accuracy*, aunque ya se ha comentado anteriormente que esta métrica no es tan significativa como el *balanced accuracy*.

[2]En negrita están los mejores resultados, por métrica y comparativa, obviando el *baseline*

TABLA II: Comparativas de los modelos de clasificación binaria contra conjunto de test

	Balanced accuracy	Accuracy
BASELINE		
Predicciones NIMA	0.9002	0.9428
Votos reales	0.6733	0.7776
LDA		
LDA	**0.7322**	**0.7512**
Predicciones NIMA	0.8791	0.8079
Votos reales	**0.7345**	0.7495
Gaussian-EM		
Gaussian-EM	0.7078	0.7361
Predicciones NIMA	**0.8995**	**0.8482**
Votos reales	0.7249	**0.7598**
K-medias		
K-medias	0.7268	0.7294
Predicciones NIMA	0.8345	0.7284
Votos reales	0.7287	0.7104
RF-IRF + LDA		
RF-IRF + LDA	0.7084	0.7268
Predicciones NIMA	0.8852	0.8169
Votos reales	0.7293	0.7487
RF-IRF suavizado + LDA		
RF-IRF suavizado + LDA	0.6994	0.7159
Predicciones NIMA	0.8778	0.8075
Votos reales	0.7313	0.7471

C. *Clasificación estética binaria*

En segundo lugar, vamos a evaluar el funcionamiento de los algoritmos de *clustering* propuestos en la Seccion III sobre el problema de **clasificación binaria**, utilizando MobileNet con salida binaria sobre las *ground-truth* calculadas por el *clustering*. Dicho *ground-truth* es la distribución de probabilidad de dos componentes obtenida por dichos algoritmos de *clustering*.

También probaremos a aplicar las transformaciones RF-IRF y su variante suavizada sobre el modelo que dé los mejores resultados de entre los modelos de *clustering*, para comprobar si pueden ayudar a estos a funcionar mejor.

Salvo por la distancia EMD media, se usarán las mismas métricas y comparaciones que en la subsección anterior.

Los resultados obtenidos por estos modelos se pueden ver en la Tabla II. Lo primero que se puede apreciar es que, mientras que todos los modelos empeoran el *accuracy* obtenido con respecto al *baseline* en todas las comparativas, también mejoran el *balanced accuracy* con respecto al *baseline*, en todas las comparativas salvo con respecto a las predicciones de NIMA. Esto se deba posiblemente a que el modelo *baseline* sea el más similar a NIMA, ya que la arquitectura de red utilizada en todos los casos es casi idéntica, y además, el modelo *baseline* ha sido entrenado con los votos sin modificar, al igual que NIMA.

De entre todos modelos, el que más destaca es **LDA**, superando en un 6% el *balanced accuracy* obtenido por el modelo *baseline*. Por otra parte, Gaussian-EM obtiene unas predicciones muy similares a NIMA, aunque aparte de obtener muy buena *accuracy* con respecto a los votos reales, no destaca tanto en el resto de resltados.

Al ser LDA el mejor modelo, se ha probado a realizar este *clustering* sobre los votos transformados según RF-IRF y su variante suavizada. Como se ve en las dos últimas secciones

de la tabla, los resultados obtenidos son sustancialmente peores con respecto a la nuevas *ground-truth* respectivas, pero también son bastante similares con respecto a los votos reales: RF-IRF suavizado + LDA está consiguiendo resultados similares a LDA con respecto a los votos reales.

Teniendo en cuenta los resultados de la Tabla I, se puede concluir que RF-IRF y su variante suavizada están consiguiendo resultados notables con respecto a los votos reales, aunque no tan notables como los de LDA. Así pues, dado que LDA y K-medias han arrojado resultados muy prometedores en cuanto a *balanced accuracy*, podrán llegar a funcionar bastante bien como nuevas *ground-truths* más sólidas que la media. También podría ser interesante utilizar RF-IRF de manera aislada como nueva *ground-truth*, dados sus buenos resultados con respecto a los votos reales.

V. Conclusiones

En este trabajo hemos propuesto varios modelos alternativos a la media para subsanar los problemas de incertidumbre que ésta presenta. En primer lugar, hemos propuesto usar algoritmos de *clustering* sobre las distribuciones de votos. En segundo lugar, hemos utilizado un modelo de pesado de calificaciones basado en TF-IDF, tanto en su versión original como en su versión logarítmica suavizada.

Para verificar su funcionamiento, hemos diseñado dos experimentos. En el primero hemos probado RF-IRF y su variante suavizada como *ground-truths* para predecir la distribución de votos, y en el segundo hemos probado los algoritmos de *clustering* como *ground-truth* para hacer clasificación binaria, probando también a aplicar RF-IRF y su variante suavizada como paso previo al *clustering* para LDA. Estos experimentos nos han permitido comprobar que RF-IRF y su variante suavizada no han obtenido los resultados esperados. No obstante, tanto LDA como K-medias han conseguido resultados significativamente superiores a NIMA, en términos de *balanced accuracy*.

Como trabajo futuro, pretendemos adaptar los mejores trabajos hasta la fecha mediante estas nuevas *ground-truth*. Para ello tenemos dos opciones: o bien cambiar las etiquetas mediante nuestras transformaciones, o bien incorporar esta nueva información a las salidas de los modelos.

VI. Agradecimientos

Este trabajo ha sido financiado parcialmente por la Agencia Estatal de Investigación (AEI), por fondos FEDER y la Junta de Comunidades de Castilla-La Mancha (JCCM) a través de los proyectos PID2019-106758GB-C33/AEI/10.13039/501100011033 y SBPLY/17/180501/000493. Luis González está además financiado por la beca 2020-PREDUCLM-14746, de la Universidad de Castilla - La Mancha (UCLM).

Referencias

[1] N. Murray, L. Marchesotti, and F. Perronnin, "Ava: A large-scale database for aesthetic visual analysis," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 2408–2415.

[2] Y. Deng, C. C. Loy, and X. Tang, "Image aesthetic assessment: An experimental survey," *CoRR*, vol. abs/1610.00838, 2016. [Online]. Available: http://arxiv.org/abs/1610.00838

[3] X. Lu, Z. L. Lin, H. Jin, J. Yang, and J. Z. Wang, "Rapid: Rating pictorial aesthetics using deep learning," in *MM '14*, 2014.

[4] L. Marchesotti, F. Perronnin, D. Larlus, and G. Csurka, "Assessing the aesthetic quality of photographs using generic image descriptors," in *2011 International Conference on Computer Vision*, 2011, pp. 1784–1791.

[5] K. Sheng, W. Dong, C. Ma, X. Mei, F. Huang, and B.-G. Hu, "Attention-based multi-patch aggregation for image aesthetic assessment," 10 2018, pp. 879–886.

[6] S. Ma, J. Liu, and C. W. Chen, "A-lamp: Adaptive layout-aware multi-patch deep convolutional neural network for photo aesthetic assessment," *CoRR*, vol. abs/1704.00248, 2017. [Online]. Available: http://arxiv.org/abs/1704.00248

[7] H. T. Esfandarani and P. Milanfar, "NIMA: neural image assessment," *CoRR*, vol. abs/1709.05424, 2017. [Online]. Available: http://arxiv.org/abs/1709.05424

[8] R. Datta, D. Joshi, J. Li, and J. Z. Wang, "Studying aesthetics in photographic images using a computational approach," in *Computer Vision – ECCV 2006*, A. Leonardis, H. Bischof, and A. Pinz, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 288–301.

[9] Y. Ke, X. Tang, and F. Jing, "The design of high-level features for photo quality assessment," in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, vol. 1. IEEE, 2006, pp. 419–426.

[10] L. Marchesotti, F. Perronnin, D. Larlus, and G. Csurka, "Assessing the aesthetic quality of photographs using generic image descriptors," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1784–1791.

[11] Yan Ke, Xiaoou Tang, and Feng Jing, "The design of high-level features for photo quality assessment," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, vol. 1, 2006, pp. 419–426.

[12] X. Tang, W. Luo, and X. Wang, "Content-based photo quality assessment," *IEEE Transactions on Multimedia*, vol. 15, no. 8, pp. 1930–1943, 2013.

[13] X. Lu, Z. Lin, H. Jin, J. Yang, and J. Z. Wang, "Rapid: Rating pictorial aesthetics using deep learning," in *Proceedings of the 22nd ACM international conference on Multimedia*. ACM, 2014, pp. 457–466.

[14] S. Kong, X. Shen, Z. Lin, R. Mech, and C. C. Fowlkes, "Photo aesthetics ranking network with attributes and content adaptation," *CoRR*, vol. abs/1606.01621, 2016. [Online]. Available: http://arxiv.org/abs/1606.01621

[15] V. Hosu, B. Goldlücke, and D. Saupe, "Effective aesthetics prediction with multi-level spatially pooled features," *CoRR*, vol. abs/1904.01382, 2019. [Online]. Available: http://arxiv.org/abs/1904.01382

[16] J. Ramos, "Using tf-idf to determine word relevance in document queries," 01 2003.

[17] S. Lloyd, "Least squares quantization in pcm," *IEEE Transactions on Information Theory*, vol. 28, no. 2, pp. 129–137, 1982.

[18] J. B. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability*, L. M. L. Cam and J. Neyman, Eds., vol. 1. University of California Press, 1967, pp. 281–297.

[19] D. Reynolds, *Gaussian Mixture Models*. Boston, MA: Springer US, 2009, pp. 659–663. [Online]. Available: https://doi.org/10.1007/978-0-387-73003-5_196

[20] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, p. 993–1022, Mar. 2003.

[21] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 39, no. 1, pp. 1–22, 1977. [Online]. Available: https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/j.2517-6161.1977.tb01600.x

[22] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

Prediction of Epiretinal Membrane from Retinal Fundus Images Using Deep Learning*

*Note: The full contents of this paper have been published in the volume Lecture Notes in Artificial Intelligence 12882 (LNAI 12882)

Ángela Casado-García, Manuel García-Domínguez, Jónathan Heras, Adrián Inés
Department of Mathematics and Computer Science
Universidad de La Rioja
La Rioja, Spain
{angela.casado,manuel.garciad,jonathan.heras,adrian.ines}@unirioja.es

Didac Royo
Optretina
Barcelona, Spain
didac@optretina.com

Miguel Ángel Zapata
Optretina
Hospital Vall Hebron
Barcelona, Spain
mazapata@optretina.com

***Abstract*—An epiretinal membrane (ERM) is an eye disease that can lead to visual distortion and, in some cases, to loss of vision. Screening retinal fundus images allows ophthalmologists to early detect and diagnose this disease; however, the manual interpretation of images is a time-consuming task. In spite of the existence of several computer vision tools for analysing retinal fundus images, they are mainly focused on the diagnosis of diabetic retinopathy and glaucoma. In this work, we have conducted a thorough study of several deep learning architectures, and a variety of techniques to train them, in order to build a model for automatically diagnosing ERM. As a result, we have built several models that can be ensembled to achieve a F1-score of 86.82%. The lessons learned in this work can serve as a basis for the construction of deep learning models for diagnosing other eye diseases.**

***Index Terms*—Epiretinal Membrane, Fundus, Image Classification, Deep Learning, Ensemble**

DOI: https://doi.org/10.1007/978-3-030-85713-4_1

Evaluation of a CNN+LSTM system for the classification of hand-washing steps

Kevin Cikel, Mario Arzamendia, Derlis Gregor
Faculty of Enginnering
National University of Asuncion
San Lorenzo, Paraguay
kcikel@ing.una.py, marzamendia@ing.una.py, dgregor@ing.una.py

Daniel Gutierrez, Sergio Toral
Escuela Técnica Superior de Ingenieros
University of Seville
Seville, Spain
dgutierrezreina@us.es, storal@us.es

***Abstract*—This paper presents the evaluation of a system based on Convolutional Neural Networks (CNN) and Long Short Term Memory (LSTM) for video classification for a small dataset. The dataset comprises 300 videos organized in 12 classes, which correspond to the 7 steps recommended by the World Health Organization (WHO) guidelines for hand-washing. We evaluated 3 models consisting of a Resnet-152 CNN encoder and a decoder based on a 3-layer LSTM, using as input the RGB frames of the videos for the first one, the optical flow for the second one, and a two-stream input made up of both RGB frames and optical flow for the third one. For the 12 original classes of the dataset, the best performance was achieved by the RGB network, which had a 78.67% categorical accuracy. By reducing the classes to 7, corresponding to the steps of the WHO hand-washing guideline, the RGB network achieved an accuracy of 97.33%.**

***Index Terms*—video classification, LSTM, convolutional neural network, hand-washing, optical flow, deep learning**

I. Introduction

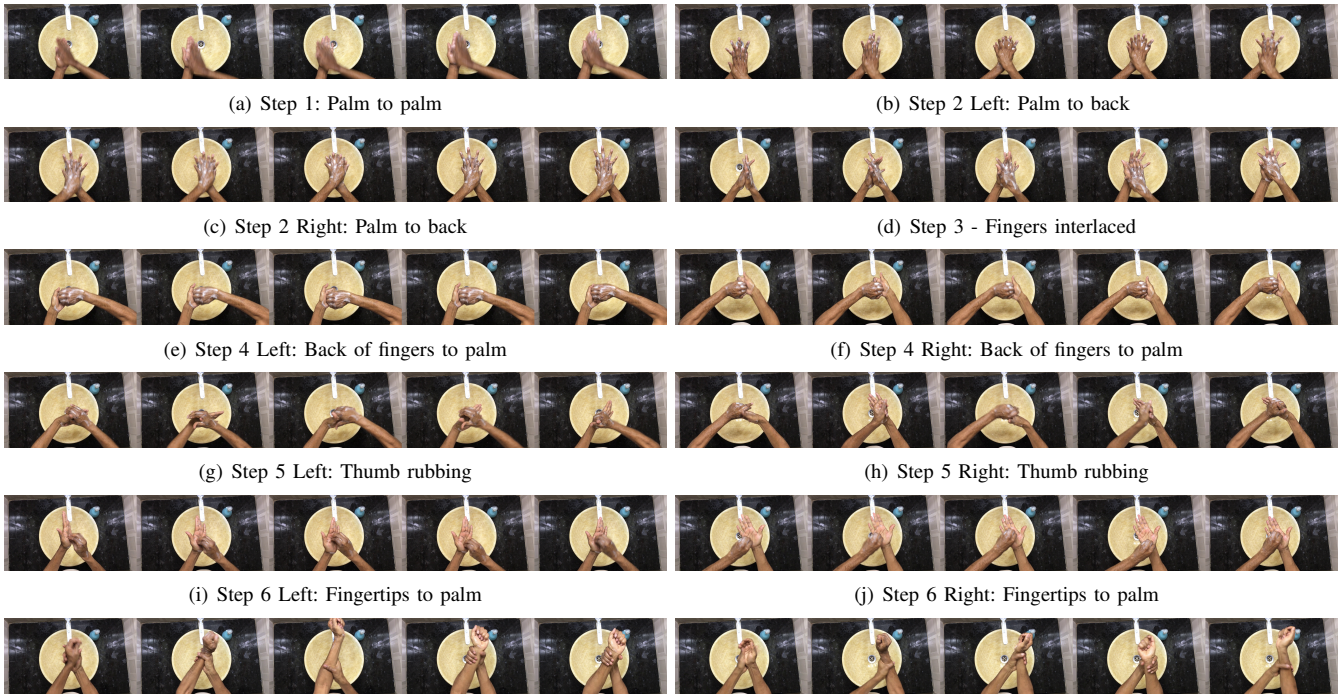
Convolutional Neural Networks (CNNs) have been used extensively for image classification, so they are also being used in the area of video classification, since a video consists of a sequence of images. Additionally, images have only spatial information, while videos also have temporal information which can be used by Recurrent Neural Networks (RNNs) such as Long Short Term Memory (LTSM). Most of the methods developed for video classification rely on CNNs to extract spatial information from video frames, while techniques such as two-stream models combining RGB frames and optical flow [1], RNNs [2] and 3D CNNs [3] have been used to deal with the temporal information. The most common applications of video classification involve action recognition, especially human actions, mainly in the form of activities, gestures and postures, for which a wide variety of datasets exist, such as the UCF101 [4] which contains 101 action classes distributed over 13,320 video clips, the HMDB51 [5] which includes 6,766 manually annotated video clips in 51 categories, and Kinetics [3] which contains 400 classes with more than 300,000 clips of real scenes extracted from Youtube. However, in this paper we deal with the Hand Wash dataset from Kaggle [6], a small dataset of only 300 video clips of hand washing steps labeled in 12 classes, according to the World Health Organization (WHO) guidelines for hand washing, which unlike the datasets mentioned above, presents very small variations in actions and environment between classes. Hand hygiene is an effective and necessary practice to combat disease transmission, and is a significantly relevant activity during the COVID-19 pandemic, where it is necessary to make it part of everyone's daily routine, hence the importance of following the guidelines for proper hand washing. Therefore, the use of deep learning to detect correct hand washing could be a valuable tool to reduce the spread of diseases. The main contribution of this paper is the evaluation of the CNN+LSTM architecture for the classification of hand washing steps in a small dataset, exploring three inputs based on RGB frames and optical flow. Because of the reduced dataset, we use a simple video classification method based on a CNN-LSTM pair, exploring different inputs such as RGB frames, optical flow and a two-stream input by combining both. The rest of the paper continues as follows, Section II presents relevant publications related to video classification and activity recognition. Section III describes the dataset and the architecture of the implemented networks. Section IV includes the results that validate the proposed approach. Finally, the paper concludes with a discussion of the results in Section V.

II. Related Works

Models based on convolutional networks has become the dominant approach for image classification problems. One such model is the residual deep network proposed in the ResNet architecture [7], which introduced a residual block to solve problem of exploding gradient, allowing to effectively train very deep network models, from hundreds to more than a thousand layers, such as the 1002-layer ResNet trained with the stochastic depth method proposed in [8]. Following the success of these networks in image classification, they have been adopted also for video classification, since videos are essentially image sequences. Many methods that utilize convolutional networks for video classification has been proposed, as in the work of Yue-Hei et al. [9], where they explored two video classification methods, the first based on temporal feature pooling and the second based on LSTM. Another method is proposed by Ullah et al. [10], where the continuous video streams are divided into important shots using a proposed CNN model, which is responsible for selecting only the important areas that are triggered by the presence

of people in a video frame. Two-stream methods are also explored, by combining the RGB frames with the optical flow of the videos, as the one presented by Simonyan et. al. [11], who proposed a two-stream CNN architecture, incorporating both spatial and temporal networks, and observed that a CNN trained on a dense optical flow of multiple frames is able to achieve very good performance in spite of limited training data, also demonstrating that multi-task learning, applied to two different action classification datasets, can be used to increase the amount of training data and improve training performance. Feichtenhofer et. al. [12] proposed a spatio-temporal fusion architecture, in where a spatial network, composed of RGB frames, and a temporal network, composed of the optical flow, are fused at the last convolutional layer by using 3D Conv fusion followed by 3D pooling. Sevilla-Lara et. al. [13] explored the integration between optical flow and action recognition, by studying the impact of different flow algorithms and input transformations on a state-of-the-art action recognition method. By performing experiments on the UCF101 action recognition dataset, they were able to make some observations: that optical flow is useful for action recognition because it is invariant to appearance; that optical flow methods are optimized to minimize the endpoint error (EPE) metric, but the EPE of current methods is not well correlated with the performance of action recognition; that improvements in performance are obtained by using optical flow to minimize classification error rather than minimizing the EPE during training; and finally, that the learned optical flow for action recognition tasks differs from traditional optical flow, especially within the human body and at body boundaries. Ma et al. [1] proposed two networks for integrating spatio-temporal information for activity recognition, one based on LTSM and the other on a temporal CNN. By training both networks with spatio-temporal feature matrices they obtained test results with the UCF101 and HMDB51 datasets of 94.1% and 69.0% accuracy respectively, which were superior to the results obtained by a baseline test performed using a two-stream CNN. To study the effectiveness of models combining a two-stream CNN with an RNN in activity recognition, Ye et al. [2] combined convolutional LTSM networks (Conv-LSTM) with a two-stream CNN. The spatio-temporal features are extracted by the pre-trained two-stream CNN, while the ConvLSTM is responsible for performing the classification of the extracted spatio-temporal features, obtaining results in experimental tests performed on the UCF101 and HMDB51 datasets of 93.9% and 69.4% accuracy, respectively. In order to deal with the problems caused by events such as occlusion and illumination changes in human activity recognition tasks, Zhu et al. [14] proposed a multimodal method in which they fuse local spatial features of video frames, global spatial features and temporal features for recognition of different actions. According to the proposed algorithm, the video is divided into three segments, the first one corresponds to an RGB frame from which the global spatial features are obtained by a spatial CNN, the second one is composed of local blocks from which the local spatial features are obtained by another spatial CNN, while the last segment corresponds to the optical flow, from which the temporal features are obtained by a temporal CNN. They evaluated the results of each of the three modes separately, as well as of different combinations of them in the UFC101 and HMDB51 datasets, obtaining the best results by using the weighted sum of the three modes. Tran et. al. [15] proposed the implementation of deep 3-dimensional convolutional network model for spatiotemporal feature learning that they called C3D (convolutional 3D).They found that the best option for 3D ConvNets is to use a homogeneous setting with convolution kernels of $3 \times 3 \times 3$. With this model they obtained 85.2% accuracy on the UCF-101 dataset when using only the RGB frames, and 90.4% accuracy when using also the optical flow and improved Dense Trajectory. Based on 2D CNN inflation, Carreira et al. [3] introduced a Two-Stream Inflated 3D CNN (I3D), transforming successful CNN models for image classification into a new 3D model by inflating all the filters and pooling kernels of the base 2D CNN model. Then, they pre-trained two networks with the Kinetics dataset, one with RGB inputs and the other with optical flow inputs. Finally, they trained both networks separately with the UCF-101 and HMDB-51 datasets and averaged their outputs, obtaining performances of 98.0% for UCF-101 and 80.9% for HMDB-51.

As for the specific application of classification of hand washing steps Zhong et. al. [16] developed a two-stage system for recognition of hand washing actions in a in a dataset composed of egocentric video obtained by wearable cameras on 100 participants, resulting in 2,055 video clips divided into 8 actions. In the first stage they extract the hand mask and motion histogram feature to localize temporal regions with potential hand washing actions, and these regions are subsequently used in the second stage as inputs to a two-stream network model to recognize the actions, achieving a detection accuracy close to 80%. A handwashing dataset that includes the steps from the WHO guidelines is presented by Lulla et al. [17], containing 3,185 videos in which each frame has been annotated as to whether it is part of the handwashing process and with its corresponding class. Eight classes are considered, in which six correspond to the steps of the WHO guidelines, one to the process of correct termination by wiping hands and closing the faucet, and the last class corresponds to any hand washing movement not considered in the previous classes. They also presented an example of application by training a model based on a MobileNetV2 CNN [18] with the dataset, considering 7 classes, obtaining an accuracy of 75.11%. As can be seen, many approaches for video classification using convolutional networks have been explored, however, most of the proposed models are evaluated on datasets with thousands of samples, so there are not many results on their performance on smaller datasets of a few hundred samples, such as the one used in this paper.



(a) Step 1: Palm to palm

(b) Step 2 Left: Palm to back

(c) Step 2 Right: Palm to back

(d) Step 3 - Fingers interlaced

(e) Step 4 Left: Back of fingers to palm

(f) Step 4 Right: Back of fingers to palm

(g) Step 5 Left: Thumb rubbing

(h) Step 5 Right: Thumb rubbing

(i) Step 6 Left: Fingertips to palm

(j) Step 6 Right: Fingertips to palm

(k) Step 7 Left: Wrist rubbing

(l) Step 7 Right: Wrist rubbing

Fig. 1. Samples from each class of the dataset.



(a)



(b)

Fig. 2. Frames of samples from the RGB dataset (a) and the optical flow dataset (b).

III. Methodology

A. Model overview

We utilized a basic CNN+LSTM model for video classification, as the one implemented in [9]. The first part of the model consists of a CNN encoder that is in charge of extracting the feature vectors from the input frames. For this we use a ResNet-152 model [7] pre-trained with the ImageNet dataset [19], that takes as input frames of 224x224 pixels. Normally very deep networks suffer from the vanishing gradient problem during training, where the gradient becomes smaller as more layers are added to the network. Residual Networks or ResNets solve this problem by the use of residual blocks, which include a connection that skips some layers, helping to overcome the vanishing gradient problem by providing an alternative route for the propagation of the gradient. This allows the training of very deep networks, such as 152 layers in this case. The second part includes a RNN decoder containing a 3-layer LSTM, which is in charge of obtaining the temporal ordering and long-term dependencies. RNNs are a type of neural networks where the output of the current input also depends on the previous outputs, so they are able to handle sequential data, while the LSTM is a special type of RNN designed to learn long-term dependencies. The RNN decoder in this part takes as input a complete sequence of activations provided by the CNN, corresponding to 128 frames for each video clip, in order to take advantage of the additional information that can be found in the variations between frames. Finnally, a fully connected layer is added to perform the classification.

B. Dataset

The dataset used is the Sample: Hand Wash Dataset from Kaggle [6], which is composed of 300 video clips, each belonging to one of 12 steps of hand-washing steps, so that each step has only 25 samples. Besides the dataset size, another difference from other commonly used datasets for video classification, such as UCF-101, HMDB-51 or Kinetics, is that the video clips in these datasets come from a large variety of actions and environments, whereas in the hand-washing dataset the differences between the classes are only in the position and movement of the hands. Due to such difference, some video classification methods that perform well on other datasets perform poorly in correctly classifying hand-washing steps. The dataset contains 7 hand washing steps based on WHO guidelines [20], which are separated into 12 classes as shown in Fig. 1. From the video clips, two datasets are generated. The first one (RGB dataset) is composed of the RGB frames, and provides the spatial features of the videos. The second one (optical flow dataset) is generated by applying the Dual TV L1 optical flow algorithm, described in [21] and [22], using the parameters from [22]. As the images in this dataset are formed from the motions in the scene, they provide the temporal features of the video. In both datasets, each sample consists of the first 128 frames of the corresponding video clip. Fig. 2 shows frame of a sample from each dataset.

C. Implementation

Three networks were trained using the CNN+LSTM model, the first with the RGB dataset and the second with the optical flow dataset. A two-stream approach was used for the third network, therefore, this network is fed with two inputs corresponding to each dataset. Then, two CNNs are used, one for each input, forwarding the feature vectors to a single LSTM decoder. The layouts of these three networks are shown in Fig. 3. The composition of the LSTM block is presentend in Fig. 4, the arrangement of the three LSTM layers is shown in (a), while (b) shows the components of a single LSTM layer, where x are the inputs coming from the output of the CNN, c are the cell states and h are the hidden states of the LSTM layers.

For training, each dataset was divided into a training set with 225 samples and a validation set with 75 samples. Also, the frames were resized to 224x224 pixels to match the ResNet-152 input.

IV. Results

For the tests we have used Python 3.7 together with the PyTorch 1.0.1 library. The training was performed on a machine with an i9-9900X CPU, 128 GB of RAM and four Nvidia GeForce RTX 2080Ti GPUs. The training time is about 490 seconds per epoch for the RGB network and the optical flow network, and 550 seconds for the two-stream network, with a total training time of about 41 hours and 46 hours respectively. For training, 300 epochs were run with a batch size of 80 and a learning rate of 0.001, also a dropout probability of 0.5 was applied to reduce overfitting.

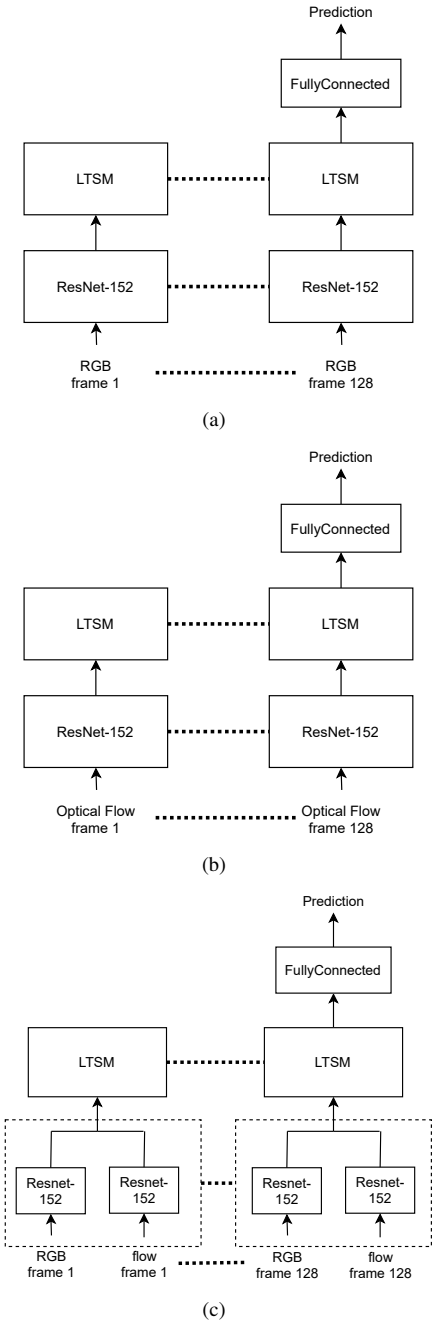


Fig. 3. Layouts of the RGB network (a), the optical flow network (b) and the two-stream network (c).

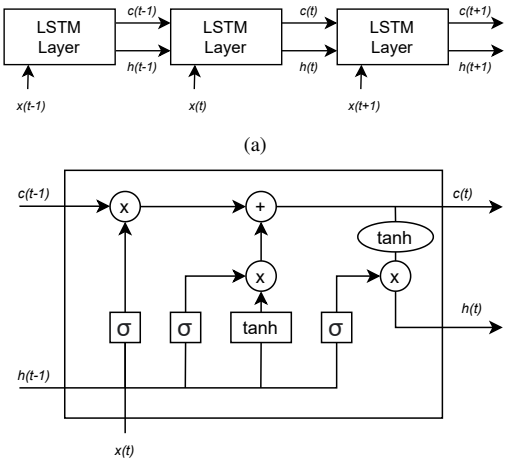


Fig. 4. Layout of the LSTM block (a) and a single LSTM layer (b).

TABLE I
TRAINING RESULTS FOR 12 CLASSES.

Input	Categorical accuracy
RGB	78.67 %
Optical flow	76.00 %
Two-stream	72.00 %

The test results are shown in the Table I. Findings reveal that the network with the best score is the one with the RGB dataset as input, which has an accuracy of 78.67%. No performance improvements were observed in the two-stream network, where the optical stream is added at the input along with the RGB frames.

The confusion matrix shown in Fig. 5 shows that the classes more frequently misclassified are 2-3, 5-6, 7-8 and 11-12, highlighted in red in the figure. Each of these pairs corresponds to a single step of the WHO guidelines for hand-washing, as shown in Figure 1: 2-3 for the palm-to-back step (Fig. 1 b and c), 5-6 for the back of fingers to palm step (Fig. 1 e and f), 7-8 for the thumb rubbing step (Fig. 1 g and h) and 11-12 for the wrist rubbing step (Fig. 1 k and l). Therefore, most of the classification errors happen between classes that belong to the same step. By performing the test again, considering now only 7 classes, each one corresponding to one step, the results shown in Table II.

TABLE II
TRAINING RESULTS FOR 7 CLASSES.

Input	Categorical accuracy
RGB	97.33 %
Optical flow	81.33 %
Two-stream	90.67 %

Fig. 5. Confusion matrix of the test results from the RGB network.

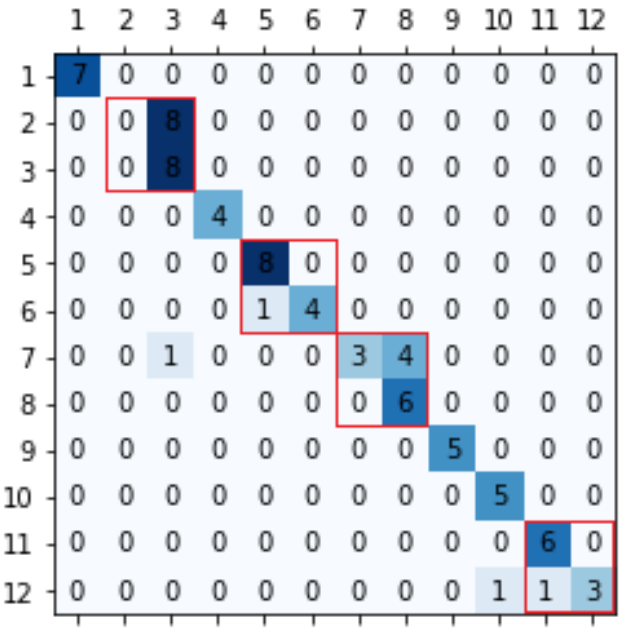
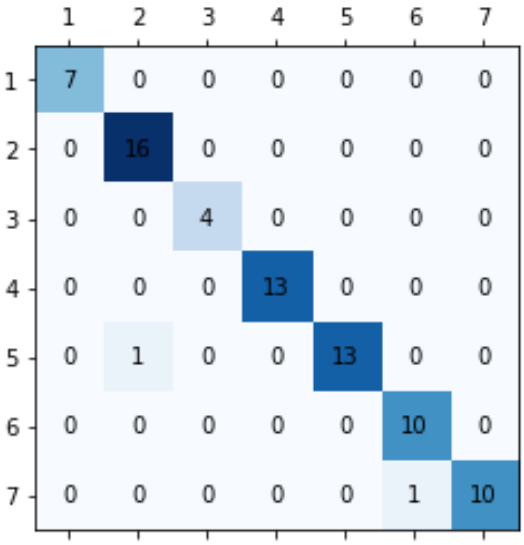


Fig. 6. Confusion matrix of the test results from the RGB network for 7 classes.



An accuracy of 97.33% was achieved for the RGB network. It can be observed that there is a notable improvement in the classification for this model and that the difference between the performance of the model trained with the RGB dataset and the other two models is now much larger. The confusion matrix of the RGB network for this test is shown in Fig. 6.

V. CONCLUSION

We evaluated the performance of a video classification method for a small dataset of hand-washing steps. From

the original dataset, an RGB frames dataset and an optical flow dataset have been generated. Three networks have been trained, including one with two-stream input, each of which it was based on a CNN ResNet-152 encoder pre-trained with the Imagenet dataset for feature vector extraction and a 3-layer LSTM decoder for sequence prediction. In the first test conducted with the original 12 classes. The RGB network produced the best result with 78.67% accuracy. A second test was performed, so that only 7 classes were considered, where the RGB network obtained an accuracy of 97.33%. As a future work remains the evaluation of the performance of other video classification models for this dataset, and also the study of the results when combining the outputs of the trained networks.

REFERENCES

[1] C.-Y. Ma, M.-H. Chen, Z. Kira, and G. AlRegib, "Ts-lstm and temporal-inception: Exploiting spatiotemporal dynamics for activity recognition," *Signal Processing: Image Communication*, vol. 71, pp. 76–87, 2019.

[2] W. Ye, J. Cheng, F. Yang, and Y. Xu, "Two-stream convolutional network for improving activity recognition using convolutional long short-term memory networks," *IEEE Access*, vol. 7, pp. 67 772–67 780, 2019.

[3] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," in *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6299–6308.

[4] K. Soomro, A. R. Zamir, and M. Shah, "Ucf101: A dataset of 101 human actions classes from videos in the wild," *arXiv preprint arXiv:1212.0402*, 2012.

[5] H. Jhuang, H. Garrote, E. Poggio, T. Serre, and T. Hmdb, "A large video database for human motion recognition," in *Proc. of IEEE International Conference on Computer Vision*, vol. 4, no. 5, 2011, p. 6.

[6] real timeAR, "Sample: Hand wash dataset," Apr 2020. [Online]. Available: https://www.kaggle.com/realtimear/hand-wash-dataset

[7] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[8] G. Huang, Y. Sun, Z. Liu, D. Sedra, and K. Q. Weinberger, "Deep networks with stochastic depth," in *European conference on computer vision*. Springer, 2016, pp. 646–661.

[9] J. Yue-Hei Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici, "Beyond short snippets: Deep networks for video classification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 4694–4702.

[10] A. Ullah, K. Muhammad, J. Del Ser, S. W. Baik, and V. H. C. de Albuquerque, "Activity recognition using temporal optical flow convolutional features and multilayer lstm," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 12, pp. 9692–9702, 2018.

[11] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Advances in neural information processing systems*, 2014, pp. 568–576.

[12] C. Feichtenhofer, A. Pinz, and A. Zisserman, "Convolutional two-stream network fusion for video action recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1933–1941.

[13] L. Sevilla-Lara, Y. Liao, F. Güney, V. Jampani, A. Geiger, and M. J. Black, "On the integration of optical flow and action recognition," in *German Conference on Pattern Recognition*. Springer, 2018, pp. 281–297.

[14] S. Zhu, Z. Fang, Y. Wang, J. Yu, and J. Du, "Multimodal activity recognition with local block cnn and attention-based spatial weighted cnn," *Journal of Visual Communication and Image Representation*, vol. 60, pp. 38–43, 2019.

[15] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 4489–4497.

[16] C. Zhong, A. R. Reibman, H. M. Cordoba, and A. J. Deering, "Hand-hygiene activity recognition in egocentric video," in *2019 IEEE 21st International Workshop on Multimedia Signal Processing (MMSP)*. IEEE, 2019, pp. 1–6.

[17] M. Lulla, A. Rutkovskis, A. Slavinska, A. Vilde, A. Gromova, M. Ivanovs, A. Skadins, R. Kadikis, and A. Elsts, "Hand-washing video dataset annotated according to the world health organization's hand-washing guidelines," *Data*, vol. 6, no. 4, p. 38, 2021.

[18] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[19] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.

[20] W. P. Safety, W. H. Organization *et al.*, "Who guidelines on hand hygiene in health care: a summary," World Health Organization, Tech. Rep., 2009.

[21] C. Zach, T. Pock, and H. Bischof, "A duality based approach for realtime tv-l 1 optical flow," in *Joint pattern recognition symposium*. Springer, 2007, pp. 214–223.

[22] J. S. Pérez, E. Meinhardt-Llopis, and G. Facciolo, "Tv-l1 optical flow estimation," *Image Processing On Line*, vol. 2013, pp. 137–150, 2013.

Ultramétricas para la comparación contextualizada de imágenes binarias

Sara Iglesias-Rey[1,3], Felipe Antunes-Santos[1,2,3], Arnau Mir-Fuentes[1,3], Carlos Lopez-Molina[1,2,3]
[1]*Dpto. Estadística, Informática y Matemáticas, Univ. Pública de Navarra, 31006 Pamplona, Spain*
[2]*KERMIT, Dept. Data Analysis and Mathematical Modelling, Ghent University, 9000 Ghent, Belgium*
[3]*NavarraBiomed, Complejo Hospitalario de Navarra, 31008 Pamplona, Spain*
sara.iglesias@unavarra.es

***Resumen*—Durante los últimos años la cuantificación de la similitud entre imágenes ha sido un tema clave en la literatura de visión artificial. En la generación de estas medidas se han utilizado diferentes teorías matemáticas, a las que nos referiremos de manera genérica como *medidas de comparación de imágenes*. El uso más frecuente de las medidas de comparación es la evaluación de resultados en diferentes tareas, aunque también tienen gran relevancia en procesos de fusión o mejora de imagen. Un aspecto interesante en el estudio de las medidas de comparación es su necesidad natural de replicar el comportamiento humano. En la mayoría de los casos, es apropiado que una medida de comparación produzca resultados coherentes con la forma en que un humano realizaría una evaluación. Sin embargo, y a pesar de aceptar esta premisa, la mayor parte de las propuestas en la literatura ignoran una característica fundamental del modo en que los humanos realizan dicha evaluación: el contexto de comparación. En este trabajo presentamos una medida de comparación para imágenes binarias que incorpora el contexto de comparación; en concreto, presentamos una metodología para la generación de ultramétricas para la comparación contextualizada de imágenes binarias. Nuestra propuesta se probará en un escenario de detección de bordes en el dataset BSDS500.**

***Palabras clave*—Comparación de imágenes, Imagen binaria, Comparación con contexto, Ultramétrica**

I. Introducción

En la literatura de procesamiento de imagen existe una gran variedad de medidas para la comparación de imágenes binarias. Esta variedad se debe principalmente a las numerosas aplicaciones de las medidas de comparación para imágenes binarias. Por ejemplo, en muchas tareas de procesamiento de imagen, los resultados intermedios o finales se representan mediante imágenes binarias, dando lugar a la necesidad de comparar las soluciones generadas de forma automática con el *ground truth*. Las medidas de comparación entre imágenes binarias serían, en este caso, las responsables directas de la evaluación de los resultados.

Las medidas de comparación de imágenes binarias surgen a partir de diversas inspiraciones. Existen ejemplos de medidas basadas en métricas, muchas de ellas extendiendo la métrica de Hausdorff [1]. También se pueden encontrar medidas de comparación basadas en funciones trigonométricas y en una interpretación vectorial de las imágenes [2]. Las medidas de comparación basadas en teoría de la información intentan modelar la información coincidente y divergente en dos imágenes binarias [3]–[5]. Esta idea se desarrolla aún más, introduciendo los conceptos de áreas de *confusión* y *consenso* en el campo de la segmentación de imágenes [4]. Aplicando medidas de comparación basadas en teoría de la información, en este caso al campo de la estadística, Fram y Deutsch [5] cuantificaron la capacidad de replicar el comportamiento humano al detectar bordes en presencia de ruido para comparar algoritmos de detección de bordes. Más allá de la amplia diversidad de medidas de comparación, cabe mencionar que la mayoría de estas medidas se presentan de forma paramétrica, dando lugar a una gama todavía más amplia de instanciaciones concretas, presentando comportamientos significativamente diferentes.

Uno de los aspectos más inexplorados en las medidas de comparación es el modelado del contexto. La mayoría de las medidas tan solo consideran las imágenes a comparar con el fin de producir una evaluación cuantitativa de la (di)similitud entre ellas. Este suele ser el caso de las métricas, que definen la distancia entre dos elementos en un universo genérico, y satisfacen los axiomas métricos correspondientes (*identidad de los indiscernibles*, *simetría* y *desigualdad triangular*).

Definición 1.1: Sea U un universo genérico, una función $d : U \times U \to [0, \infty[$ se denomina una métrica en U si satisface las siguiente propiedades:

1. *Identidad de los indiscernibles*: $d(x, y) = 0$ si y solo si $x = y$.
2. *Simetría:* $d(x, y) = d(y, x)$ para cualquier $x, y \in U$.
3. *Desigualdad triangular:* $d(x, z) \leq d(x, y) + d(y, z)$ para cualquier $x, y, z \in U$.

Las métricas son las funciones más usadas para la comparación de datos, si bien existen muchas otras. Por ejemplo, si en lugar de la desigualdad triangular, la función d satisface la *desigualdad ultramétrica* o de *triángulo fuerte*, $d(x, y) \leq \max(d(x, z), d(y, z))$ para cualquier $x, y, z \in U$, la función resultante sería una ultramétrica. Obviamente, cualquier ultramétrica es una métrica.

Nótese que una métrica es una ultramétrica [6] si y solo si tres puntos cualquiera en U pueden ser reetiquetados como x, y, z tal que

$$d(x, y) \leq d(x, z) = d(y, z)\,.$$

El uso de métricas para comparación de imágenes puede parecer un enfoque natural y matemáticamente adecuado, pero de hecho es diferente a cómo proceden los humanos. Los seres

humanos incorporan de forma implícita una contextualización en sus comparaciones, realizando un análisis multidimensional de las características de cada objeto. Esta incorporación del contexto puede incluso dar lugar a que no se cumplan de forma natural los axiomas al realizar la comparación. El contexto puede ser explícito, pero si no se proporciona ninguna información, los humanos lo incorporarán de forma implícita.

Un ejemplo ilustrativo de esta contextualización implícita realizada por los humanos es la comparación *Jamaica-Cuba-Rusia*, presentada por Tversky [7]. El ejemplo se basa en la idea de que, siendo d una medida de disimilitud entre países, los humanos normalmente juzgan que $d(\textit{Cuba}, \textit{Jamaica}) + d(\textit{Cuba}, \textit{Russia}) < d(\textit{Jamaica}, \textit{Russia})$. Dicho de otro modo, que la suma de la disimilitud por pares entre *Jamaica* y *Cuba*, y entre *Cuba* y *Rusia*, es menor que la disimilitud percibida entre *Jamaica* y *Rusia*, rompiendo así la desigualdad triangular. La razón radica en que cada país es considerado un objeto multidimensional y, que en la comparación de cada par de países, los humanos determinan implícitamente la dimensión de comparación, es decir, el contexto de comparación. Incluso cuando el contexto no es explícito, el evaluador humano lo determina de forma implícita según experiencias previas. Por ejemplo, la comparación *Jamaica-Cuba* se establece normalmente en términos de geografía, mientras que *Cuba-Rusia* se basa en su política; los seres humanos alteran selectivamente el papel y la configuración del contexto en las comparaciones. Este ejemplo se utiliza para desacreditar la necesidad de imponer la desigualdad triangular en el pensamiento humano, así como para arrojar luz sobre la naturaleza multidimensional de la interpretación humana. Además sirve como ilustración del interesante rol del contexto en las comparaciones humanas [8].

En general, las medidas de comparación en la literatura no hacen uso de la noción de contexto. Normalmente, sólo consideran las dos imágenes que se van a comparar, asumiendo un contexto global y, por tanto, proporcionando una cuantificación absoluta de (di)similitud. Si bien esto puede ser ventajoso en algunas tareas, el proceso se aleja de la forma en la que los humanos realizan comparaciones. El objetivo de este trabajo es presentar una medida de comparación de imágenes binarias en la que el contexto juega un papel activo. De forma específica, crearemos una ultramétrica que cuantifique la disimilitud entre dos imágenes cualesquiera dentro del contexto de comparación. La distancia cuantificada entre pares de imágenes se obtiene no solo en términos de sus coincidencias y divergencias, sino también en base a las características de las imágenes restantes dentro del contexto de comparación. Mediante la inclusión de un contexto, nuestra medida acerca el proceso a cómo los humanos realizan comparaciones de forma natural.

El resto de este artículo se organiza de la siguiente manera. La Sección II presenta la importancia de las medidas de comparación sensibles al contexto. La metodología para la construcción de ultramétricas se presenta en la Sección III, y se evaluará en la Sección IV. Finalmente, la Sección V presenta las conclusiones del artículo.

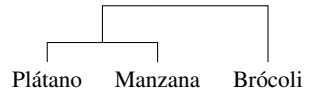
II. Ultramétricas para la comparación de objetos

II-A. Clases de medidas de comparación

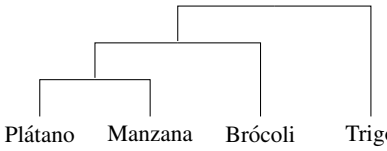
Las medidas de comparación, ya sean métricas, ultramétricas o cualquier otro tipo de función, son cruciales en la mayoría de las áreas científicas, siendo decisivas en tareas como evaluación de la calidad, optimización o *clustering*. En términos de Tversky [9], las medidas de comparación se pueden dividir en dos clases principales: las basadas en interpretaciones geométricas, llamadas *modelos espaciales*, y las basadas en la teoría de grafos, llamadas *modelos de red*. Los modelos espaciales representan cada objeto como un punto en un espacio de coordenadas, de modo que la distancia entre puntos representa la proximidad entre objetos. La mayoría de las métricas y medidas de disimilitud se adhieren a esta estrategia, incluidas, por ejemplo, las diferentes extensiones de la métrica de Hausdorff [10], [11]. Como alternativa a los modelos espaciales, los modelos de red generan una representación de grafos de las relaciones entre los objetos a comparar. Cada objeto se define como un nodo en un grafo conectado y acíclico, generalmente un árbol binario enraizado y etiquetado, mientras que las *aristas* o *ramas* (y sus pesos) se utilizan para representar la disimilitud (o distancia) entre cada par de objetos. Cuando se usan árboles para generar estructuras jerárquicas, los nodos de grado 1 se denominan *hojas* y el nodo de grado 2 *raíz*.

En la doble taxonomía de Tversky, los modelos de red son más adecuados para modelar la noción de contexto que los modelos espaciales. Los modelos espaciales pueden establecer de forma potencial un espacio de coordenadas dependiendo de alguna definición del contexto; sin embargo, esto se encuentra ausente en la literatura de procesamiento de imágenes, excepto en aplicaciones concretas de consulta de imágenes mediante comparación de distribuciones [12]. Generalmente se asume que el contexto es lo suficientemente grande e inespecífico, y que contiene *todos los elementos posibles*. Los modelos de red permiten una fácil adaptación al contexto al establecer la topología del grafo, ya que se consideran todas las posibles interrelaciones entre elementos. Por lo tanto, el conjunto de objetos a comparar afecta a la topología de la red y, con ello, a la cuantificación de la disimilitud entre dos nodos cualquiera.

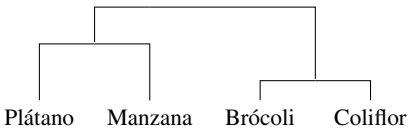
La generación de un grafo, sobre el que se construye una ultramétrica, permite modelar las relaciones entre imágenes de un conjunto. Esta alternativa posibilita un mejor modelado del comportamiento humano al comparar y evaluar la disimilitud, ya que incluye de forma implícita el contexto. Además, al satisfacer las propiedades de la ultramétrica, garantizamos que nuestra medida de comparación se trata de una métrica y, por lo tanto, puede ser utilizada en distintas aplicaciones que precisen métricas. El principal inconveniente de esta alternativa es que la topología de la red no puede establecer distancias desde o hacia elementos que no estaban presentes cuando se definió la estructura, por lo que todos los elementos a comparar deben conocerse de antemano.



(a) Modelo de árbol de {Plátano, Manzana, Brócoli}.



(b) Modelo de árbol de {Plátano, Manzana, Brócoli}∪{Trigo}.



(c) Modelo de árbol de {Plátano, Manzana, Brócoli}∪{Coliflor}.

Figura 1: Modelo de árbol para la representación de similitud entre términos lingüísticos. Se presentan tres variantes del conjunto de términos.

II-B. Comparación basada en modelos de red

Las medidas de comparación basadas en redes se obtienen estableciendo una topología de red sobre un conjunto de elementos. Aunque esta topología de red puede ser cualquier tipo de grafo (las distancias en los mapas de carreteras cíclicas son un ejemplo de ello), generalmente los árboles enraizados son la opción predilecta, ya que imponen una estructura jerárquica adecuada sobre los datos. Esta elección no solo es relevante en el contexto actual, sino que también es fundamental, por ejemplo, en algoritmos de *clustering* jerárquico. Este trabajo se centra en el uso de árboles como topologías de red.

La Figura 1 muestra un ejemplo de una organización en forma de árbol de términos lingüísticos basada en la disimilitud. Este grafo contiene, como hojas, los términos iniciales Plátano, Manzana y Brócoli, y los agrupamientos resultantes de los elementos. En la Fig. 1(a), {Plátano, Manzana} son los primeros agrupados, ya que son los términos más cercanos en el conjunto. A continuación, {Plátano, Manzana} se agrupan con {Brócoli}, formando la raíz del árbol. Por ahora evitaremos detalles numéricos, por lo que asumimos que la disimilitud entre dos elementos es proporcional a la altura del nodo en el que se agrupan por primera vez.

Las Figuras 1(b) y 1(c) muestran los árboles resultantes al agregar dos términos distintos al conjunto de candidatos de la Fig. 1(a): Trigo y Coliflor. En el primer caso, al incluir un elemento perceptualmente distante de todos los elementos del conjunto original, Trigo, la topología del árbol no cambia de forma significativa. En la Fig. 1(c), encontramos una situación diferente. Al incluir el elemento Coliflor, que se percibe como cercano a uno de los elementos del conjunto original, Brócoli, el árbol cambia significativamente. Esto representa el hecho de que los humanos entienden que los términos son similares de forma variable, dependiendo del contexto de comparación.

II-C. Medidas de comparación basadas en árboles

Los ejemplos de la Fig. 1 proporcionan una idea de cómo organizar los términos de acuerdo a su disimilitud percibida, pero no explican cómo generar ni una ultramétrica a partir de ellos, ni cualquier otra medida de comparación. El proceso de construcción del árbol comienza considerando todos los objetos que se van a comparar y estableciendo el primer nodo combinado como el *par más cercano* de términos entre todos los pares candidatos. Esta elección no es trivial; en términos computacionales, múltiples paradigmas distintos pueden estar involucrados: entropía mínima del conjunto resultante, menor disimilitud usando una medida o métrica, probabilidad condicional, etc. Es necesario decidir de forma iterativa cuál es el siguiente nodo a generar, hasta que la raíz sea alcanzada. Nótese que la estrategia usada para seleccionar el *siguiente nodo a combinar* no debe únicamente considerar el agrupamiento de nodos hoja; si no que también debe estar preparada para considerar los nodos intermedios que se generen en el árbol, hasta que se alcance la raíz.

Una vez la taxonomía del árbol está establecida, es importante definir pesos para las aristas o ramas. Dependiendo de la naturaleza de la estrategia utilizada para la fusión de nodos, podemos llegar a dos situaciones diferentes. Si el enfoque es cualitativo, *i.e.*, determina cuál es el siguiente nodo que se generará pero no arroja una evaluación numérica de su coste, cada nodo nuevo se sitúa a una altura creciente en el árbol. El primer nodo creado combinando los elementos de las hojas tendrá un coste de 1, el segundo tendrá un coste de 2 y así sucesivamente. Si la estrategia utilizada para la agrupación de nodos es de naturaleza numérica, se considera que cada nodo nuevo tiene una altura equivalente al coste de su generación. Téngase en cuenta que esta altura no es una propiedad intrínseca de los nodos, si no que está representada por la longitud de las aristas que conectan cada nodo a las hojas.

Las estrategias cuantitativas deben cumplir ciertas propiedades que garanticen, por ejemplo, que los nodos se crean con una altura creciente, de forma que cada nodo nuevo no puede tener un coste menor que los nodos más antiguos. También se debe considerar la situación en la que diferentes nodos candidatos presenten el mismo coste durante el proceso de creación del árbol; existen múltiples estrategias diferentes para superar esta situación como crear varios nodos con el mismo coste simultáneamente o una elección aleatoria.

Cualquiera que sea la estrategia empleada para diseñar el árbol, el grafo se puede utilizar para crear una ultramétrica. La distancia entre dos elementos cualesquiera en el conjunto de elementos originales, en términos de la ultramétrica, es la altura del nodo más bajo que incluye a ambos. Esto concuerda con la propiedad de *triángulo fuerte* en Def. 1.1 ya que la distancia entre dos elementos cumple $d(x,y) \leq \max(d(x,z), d(y,z))$ para cualquier $x, y, z \in U$. Esta medida dependerá del contexto de comparación, ya que diferentes grupos de elementos darán lugar a distintas topologías de árbol y, por lo tanto, la ultramétrica arrojará diferentes valores.

III. Ultramétricas para la comparación de imágenes binarias

III-A. Comparación de imágenes binarias

El uso extendido de imágenes binarias para la representación de resultados intermedios (o finales) en procesamiento de imagen dio lugar a la proliferación de una gran variedad de medidas para la comparación de imágenes binarias. Gran parte de la literatura está dedicada a medidas basadas en matrices de confusión [3], [4],pero también existe un número significativo de medidas que adoptan una interpretación geométrica del espacio de imágenes binarias [2]. Las opciones más populares en esta dirección son la métrica de Hausdorff [1] y diferentes generalizaciones orientadas a tareas específicas [10], [11] o la Diferencia Simétrica [13]. Otras medidas, aunque no satisfacen los axiomas métricos, se basan parcialmente en métricas. Algunos ejemplos son el FoM de Pratt (PFoM) [14] y la medida de Haralick [15].

Todas estas medidas cumplen determinadas propiedades, axiomas métricos o no, pero no ofrecen una forma fácil e intuitiva de incorporar el contexto de comparación. Hasta donde sabemos, prácticamente no existen referencias en la literatura que presenten medidas de comparación sensibles al contexto, donde cada comparación se realice teniendo en cuenta todas las imágenes binarias dentro de un conjunto. De hecho, cuando se usa, el término *contexto* se relaciona con la idea de que diferentes regiones de la imagen presentan entornos locales y globales discriminatorios, proporcionando una medida de comparación adaptativa para cada imagen; por tanto, el contexto se refiere a las relaciones de un píxel con su vecindad. Se pueden encontrar ejemplos en una gran variedad de aplicaciones como *matching* de formas [16] o bordes [17], o en análisis multiescala de mapas de superpíxeles [18].

Las únicas medidas que consideran la noción de contexto como se presenta en este trabajo, se centran en la comparación de conceptos. La mayoría de estas medidas surgen como una noción fundamental de las teorías cognitivas en psicología [19]. Por ejemplo, al analizar las frecuencias de co-ocurrencia entre pares de palabras en alemán, es posible simular una medida de disimilitud semántica dentro de un contexto [20]. Otros ejemplos de análisis del contexto en la literatura se centran en modelos en teoría de la elección dependientes del contexto [21], donde Tversky incorpora diferentes pesos para simular el efecto del fondo y el contexto local. A pesar de considerar la noción de contexto como se presenta en este documento, estas medidas solo se aplican para la comparación de conceptos. No existen ejemplos en la literatura de procesamiento de imágenes en los que se intente incluir nuestra noción de contexto a la comparación de imágenes binarias.

Proponemos una medida de comparación sensible al contexto para imágenes binarias, utilizando árboles para obtener una cuantificación de disimilitud por medio de una ultramétrica.

III-B. Comparación basada en árboles de imágenes binarias

La construcción de la ultramétrica basada en árboles presentada en la Sección II puede ser extendida al caso de las imágenes binarias. Para ello se requiere una herramienta matemática que permita el agrupamiento en nodos de objetos, es decir, un método cuantitativo para llevar a cabo comparaciones entre imágenes. Debido a la naturaleza de los árboles jerárquicos, esta herramienta se utilizará para decidir el orden en el cual diferentes imágenes se agruparán en los nodos. En nuestra propuesta, las comparaciones entre imágenes se realizarán usando la Métrica Delta de Baddeley (BDM por sus siglas en inglés) [22], [23] una medida de comparación muy popular derivada de la métrica de Hausdorff [1].

En este trabajo consideramos que las imágenes tienen unas dimensiones fijas $\mathcal{M} \times \mathcal{N}$, tal que $\Omega = \{1, \ldots, \mathcal{M}\} \times \{1, \ldots, \mathcal{N}\}$ representa el conjunto de posiciones en la imagen. El conjunto de todas las imágenes binarias se denomina como $\mathbb{B}$, *i.e.*, el conjunto de mapeos $\Omega \mapsto \{0, 1\}$.

Sean $A, B \in \mathbb{B}$ dos imágenes binarias en Ω y m una métrica en Ω. La distancia entre ellas, en términos de BDM viene dada por

$$\Delta^k(A, B) = \left[\frac{1}{|\Omega|} \sum_{p \in \Omega} |w(\mathcal{T}_m[A](p)) - w(\mathcal{T}_m[B](p))|^k \right]^{\frac{1}{k}}, \tag{1}$$

donde $w : \mathbb{R}^+ \mapsto \mathbb{R}^+$ es una función cóncava con $w(x) = 0$ si y solo si $x = 0$, $k \in \mathbb{R}^+$, y $\mathcal{T}_m$ es una transformada de distancia de imagen definida como

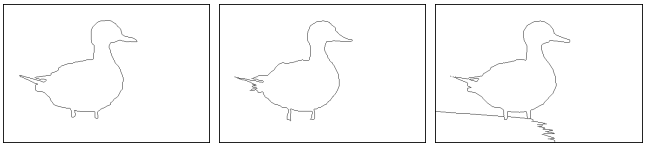
$$\mathcal{T}_m[I](p) = \min_{p' \in I} m(p, p') , \tag{2}$$

para todo $p \in \Omega$.

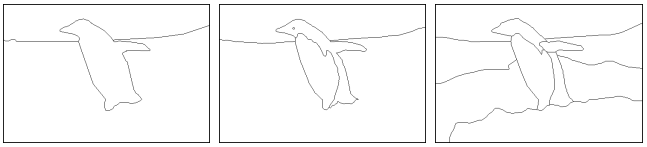
Esta medida proporciona una cuantificación de la distancia entre imágenes en el conjunto y se trata de una estrategia válida para diseñar una medida de disimilitud jerárquica basada en árboles, obteniendo una relación de proporcionalidad entre las distancias entre elementos y la topología del árbol. También permite la creación de una ultramétrica, ya que la distancia entre dos imágenes en el conjunto es la altura del nodo más bajo que incluye a ambas. Este enfoque proporciona una representación gráfica e intuitiva de la proximidad de los elementos a comparar del conjunto original. La adición de nuevas imágenes al conjunto original dará lugar a diferentes cuantificaciones de la disimilitud y, en consecuencia, a diferentes configuraciones del árbol ultramétrico, debido a su sensibilidad al contexto.

La construcción de árboles ultramétricos requiere la comparación no solo de imágenes, sino de conjuntos de imágenes en los nodos superiores. Sin embargo, muy pocas medidas en la literatura permiten la comparación N-a-M de imágenes binarias. Las medidas de comparación 1-a-1 se pueden utilizar para la generación de medidas de comparación N-a-M. A partir de BDM podemos obtener una medida de comparación entre conjuntos de imágenes binarias **A**,**B** tal que

$$C(\mathbf{A}, \mathbf{B}) = \bigwedge_{i,j} \Delta^k(A_i, B_j) . \tag{3}$$



(a) *Ground truth* asociados a la imagen 43051 del BSDS500



(b) *Ground truth* asociados a la imagen 106005 del BSDS500

Figura 2: Imágenes *ground truth* de bordes del dataset BSDS500. El etiquetado por parte de varios humanos da lugar a una alta variabilidad entre imágenes pertenecientes a la misma clase.

IV. Experimentos

IV-A. Datos experimentales

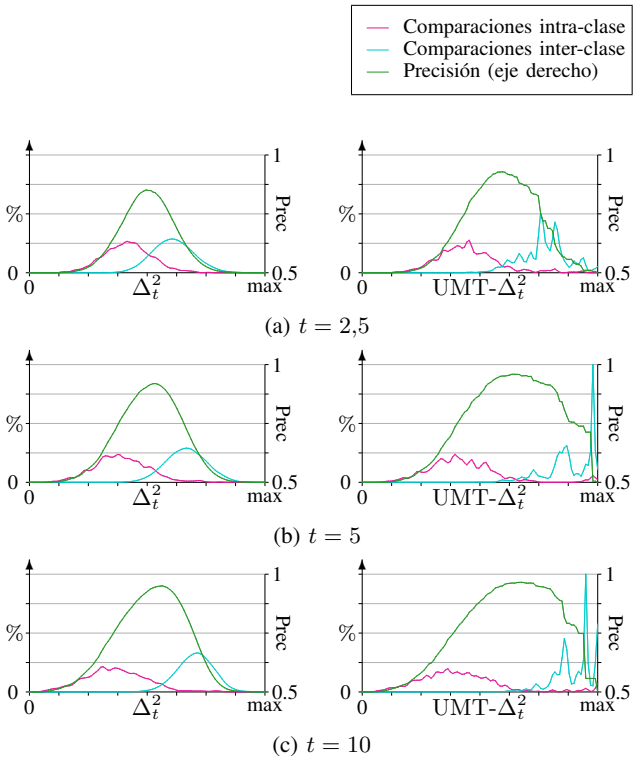
El *Berkeley Segmentation Data Set and Benchmark 500* (BSDS500) [24] es un dataset muy popular para aplicaciones de detección de bordes y segmentación de imágenes. Contiene un gran conjunto de imágenes, cada una asociada a una colección de imágenes *ground truth* de bordes etiquetadas a mano. La Fig. 2 muestra imágenes *ground truth* correspondientes a dos imágenes del dataset BSDS500 pertenecientes a clases distintas (asociadas a distintos *ground truth*).

A pesar de que los humanos incurren en una gran variabilidad al etiquetar imágenes, cualquier humano puede de forma general agrupar las imágenes que proceden de la misma imagen original (intra-clase), y diferenciarlas de las que no (inter-clase). Cualquier medida de comparación debería poder replicar este comportamiento humano, produciendo valores más bajos para imágenes intra-clase que para las inter-clase. En la Sección IV-B evaluaremos la capacidad de discriminar entre pares de imágenes intra-clase e inter-clase de dos medidas de comparación distintas basadas en BDM.

IV-B. Análisis de separabilidad

La evaluación, medición o comparación del rendimiento de medidas de comparación no son triviales. En este trabajo analizamos si diferentes medidas de comparación para imágenes binarias son capaces de discriminar cuando dos imágenes de bordes en el BSDS500 se generan a partir de la misma imagen original o no; es decir, si la medida de comparación genera mayores valores de disimilitud para las comparaciones inter-clase que para las comparaciones intra-clase. Se considera que un par de imágenes es intra-clase si están generadas (por diferentes humanos) a partir de la misma imagen original, e inter-clase si no lo son. Dado que esta es una tarea que los humanos pueden realizar con facilidad, puede usarse como una medida simple de la replicación del comportamiento humano.

En este experimento, hemos realizado la comparación de todos los posibles pares de imágenes usando diferentes BDMs



(a) $t = 2{,}5$

(b) $t = 5$

(c) $t = 10$

Figura 3: Distribuciones intra- e inter-clase de los valores obtenidos mediante Δ_t^2 y UMT-Δ_t^2, $t \in \{2{,}5, 5, 1\}$, en el las imágenes del dataset BSDS500 [24]. Se incluye igualmente la precisión en las separación de ambas por cada umbra en el intervalo. Las distribuciones se configuran con 100 bins.

y árboles ultramétricos a partir de estas medidas. Para estos experimentos, fijamos $w(x) = x$, $k = 2$, y m la distancia euclídea acotada $m_t(p, p') = \min(t, m(p, p'))$ para cualquier $p, p' \in \Omega$, con $t = \{2{,}5, 5, 10\}$; nos referiremos a las medidas directas y a las generadas con árboles ultramétricos como Δ_t^2 y UMT-Δ_t^2, respectivamente. El dataset BSDS500 Test contiene 200 clases distintas, dando lugar a 1063 imágenes en total. Esto implica que el número de comparaciones inter-clase es varios órdenes de magnitud mayor que el de comparaciones intra-clase. También calculamos la precisión esperada (Prec) de discriminación de las distribuciones para cada posible umbral. Idealmente, $\mathrm{Prec} = 1$ para al menos un umbral, si ambas distribuciones no se superponen, lo que produce una separabilidad total.

En la Figura 3 presentamos las distribuciones de las comparaciones intra- e inter-clase de las medidas de comparación directas Δ_t^2 (columna izquierda) y las medidas de comparación basadas en árboles ultramétricos UMT-Δ_t^2 (columna derecha). Además, se señala la precisión como índice de separación de las dos clases para un umbral dado. Nótese que si las distribuciones fueran disjuntas, la precisión alcanzaría el valor 1. Replicamos los resultados con $t \in \{2{,}5, 5, 10\}$. Es relevante señalar que, debido a la gran diferencia entre el número de

comparaciones intra-clase e inter-clase, las distribuciones se presentan en términos porcentuales. Nótese también que las distribuciones se muestran en el eje izquierdo, mientras que Prec se mide en el eje derecho, tomando valores entre $0{,}5$ y 1.

En general, las distribuciones de las comparaciones intra-clase e inter-clase en la Fig. 3 son bastante separables para todas las medidas de comparación, usando árboles o no. Las comparaciones intra-clase normalmente dan lugar a valores menores que las comparaciones inter-clase para todas las configuraciones, replicando de forma correcta el comportamiento humano. Sin embargo, se puede observar a primera vista que algunas medidas de comparación producen distribuciones intra-clase e inter-clase que, visualmente, son más distantes que otras.

Centrándonos en la comparación entre medidas de comparación estándar (directas) y basadas en árboles, podemos ver hechos muy interesantes. A primera vista, podemos observar que UMT-Δ_t^2 presenta una mejora notable respecto a Δ_t^2 para todos los valores de t. Las medidas de comparación directas presentan una separabilidad mucho menor entre las distribuciones intra-clase e inter-clase; por el contrario las configuraciones basadas en árboles presentan no solo picos de Prec más elevados, sino también un área mayor bajo la curva. Por ello, podemos afirmar que el uso de árboles da lugar no solo a una mejora cualitativa mediante la inclusión del contexto de comparación, sino que también se obtiene una mejora cuantitativa de la separabilidad.

V. Conclusiones

En este trabajo presentamos una medida de comparación sensible al contexto basada en árboles ultramétricos para imágenes binarias. Para realizar las cuantificaciones de comparación entre imágenes binarias, aplicamos BDM [22], [23]. Aplicamos nuestro algoritmo al dataset BSDS500 y realizamos un análisis de separabilidad, obteniendo la precisión y las distribuciones correspondientes.

Como conclusión, podemos afirmar que la construcción de un árbol ultramétrico aplicando nociones matemáticas simples, permite modelar el contexto en la comparación de imágenes binarias. Sin embargo, esta construcción debe estar respaldada por una medida de comparación que proporcione una cuantificación para la selección del nodo. Mediante el uso de BDM podemos afirmar que nuestro algoritmo no solo proporciona una medida de comparación sensible al contexto, sino que también una mejor reproducción del comportamiento humano.

Agradecimientos

Los autores agradecen el soporte financiero del Ministerio Español de ciencia (proyecto PID2019-108392GB-I00 AEI/FEDER, UE), así como el de Navarra de Servicios y Tecnologías, S.A. (NASERTIC). También la financiación del programa de investigación e innovación de la Unión Europea H2020 bajo la beca Marie Sklodowska-Curie Número 801586.

Referencias

[1] D.-G. Sim, O.-K. Kwon, and R.-H. Park, "Object matching algorithms using robust Hausdorff distance measures," *IEEE Trans. on Image Processing*, vol. 8, no. 3, pp. 425–429, 1999.

[2] J. Gimenez, J. Martinez, and A. G. Flesia, "Unsupervised edge map scoring: A statistical complexity approach," *Computer Vision and Image Understanding*, vol. 122, pp. 131–142, 2014.

[3] S. K. Warfield, K. H. Zou, and W. M. Wells, "Simultaneous truth and performance level estimation (STAPLE): An algorithm for the validation of image segmentation," *IEEE Trans. on Medical Imaging*, vol. 23, no. 7, pp. 903–921, 2004.

[4] A. Asman and B. Landman, "Robust statistical label fusion through consensus level, labeler accuracy and truth estimation (COLLATE)," *IEEE Trans. on Medical Imaging*, vol. 30, pp. 1779–94, 04 2011.

[5] J. Fram and E. S. Deutsch, "On the quantitative evaluation of edge detection schemes and their comparison with human performance," *IEEE Trans. on Computers*, vol. 24, no. 6, pp. 616–628, 1975.

[6] T. Faver, K. Kochalski, M. K. Murugan, H. Verheggen, E. Wesson, and A. Weston, "Roundness properties of ultrametric spaces," *Glasgow Mathematical Journal*, vol. 56, no. 3, pp. 519–535, 2014.

[7] A. Tversky, "Features of similarity," *Psychological Review*, vol. 84, no. 4, pp. 327–352, 1977.

[8] J. Fumanal-Idocin, A. Alonso-Betanzos, O. Cordón, H. Bustince, and M. Minárová, "Community detection and social network analysis based on the Italian Wars of the 15th century," *Future Generation Computer Systems*, vol. 113, pp. 25–40, 2020.

[9] A. Tversky, *Preference, belief, and similarity : selected writings by Amos Tversky; edited by Eldar Shafir*. Cambridge, Mass: MIT Press, 2004.

[10] M. Dubuisson and A. K. Jain, "A modified Hausdorff distance for object matching," in *Proc. on Pattern Recognition*, vol. 1, 1994, pp. 566–568.

[11] B. Takács, "Comparing face images using the modified Hausdorff distance," *Pattern Recognition*, vol. 31, no. 12, pp. 1873–1881, 1998.

[12] F. Perronnin, Y. Liu, and J.-M. Renders, "A family of contextual measures of similarity between distributions with application to image retrieval," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 2358–2365.

[13] C. Lopez-Molina, B. De Baets, and H. Bustince, "Quantitative error measures for edge detection," *Pattern Recognition*, vol. 46, no. 4, pp. 1125–1139, 2013.

[14] I. Abdou and W. Pratt, "Quantitative design and evaluation of enhancement/thresholding edge detectors," in *Proc. of the IEEE*, vol. 67, 1979, pp. 753–763.

[15] R. M. Haralick, "Digital step edges from zero crossing of second directional derivatives," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 6, no. 1, pp. 58–68, 1984.

[16] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, no. 4, pp. 509–522, 2002.

[17] Y.-H. Yu and C.-C. Chang, "A new edge detection approach based on image context analysis," *Image and Vision Computing*, vol. 24, no. 10, pp. 1090–1102, 2006.

[18] H. Jiang, J. Wang, Z. Yuan, T. Liu, N. Zheng, and S. Li, "Automatic salient object segmentation based on context and shape prior," in *Proc. British Machine Vision Conference (BMVC)*, 2011, pp. 110.1–110.12.

[19] M. Bazire and P. Brézillon, "Understanding context before using it," in *Modeling and Using Context*, A. Dey, B. Kokinov, D. Leake, and R. Turner, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 29–40.

[20] S. Bordag, "A comparison of co-occurrence and similarity measures as simulations of context," in *Computational Linguistics and Intelligent Text Processing*, A. Gelbukh, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 52–63.

[21] A. Tversky and I. Simonson, "Context-dependent preferences," *Management Science*, vol. 39, no. 10, pp. 1179–1189, 1993.

[22] A. J. Baddeley, "An error metric for binary images," in *Robust Computer Vision: Quality of Vision Algorithms*, W. Förstner and S. Ruwiedel, Eds. Karlsruhe: Wichmann Verlag, 1992, pp. 59–78.

[23] ——, "Errors in binary images and an L^p version of the Hausdorff metric," *Nieuw Archief voor Wiskunde*, vol. 10, pp. 157–183, 1992.

[24] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proc. of the IEEE International Conf. on Computer Vision*, vol. 2, 2001, pp. 416–423.

Towards Combining Object Detection and Text Classification Models for Form Entity Recognition*

*Note: The full contents of this paper have been published in the volume Lecture Notes in Artificial Intelligence 12882 (LNAI 12882)

María Villota, Gonzalo Santamaría, César Domínguez, Jónathan Heras, Eloy Mata, Vico Pascual
Department of Mathematics and Computer Science
Universidad de La Rioja
La Rioja, Spain
{maria.villota,gonzalo.santamaria,cesar.dominguez,jonathan.heras,eloy.mata,vico.pascual}@unirioja.es

***Abstract*—Due to the increasing volume of forms that are generated in a daily basis, the automatic extraction of the information included in these template-based documents is greatly demanded. However, this is not a straightforward task due to the great diversity of templates with different location of form entities, and the quality of the scanned documents. In this work, we have made a first step towards form entity recognition by combining computer vision and natural language processing techniques. First, we have applied state-of-the-art deep object detection and semantic segmentation models for localising the position of form entities based only on visual features. Afterwards, we have studied different transfer learning approaches (fine-tuning and feature extraction) for classifying the text content of the localised form entities. The studied models require low computational and image resources, making them a feasible alternative to state-of-the-art models even if their performance is slightly worse.**

***Index Terms*—Form Understanding, Object Detection, Semantic Segmentation, Text Classification.**

DOI: https://doi.org/10.1007/978-3-030-85713-4_19

AutoML para la composición de workflows diversos mediante programación genética gramatical

Rafael Barbudo
Dept. de Informática y Análisis Numérico
Universidad de Córdoba, Córdoba 14071
Email: rbarbudo@uco.es

Sebastián Ventura y José Raúl Romero
Dept. de Informática y Análisis Numérico,
Instituto Interuniversitario de Investigación en
Data Science and Computational Intelligence (DaCSI)
Universidad de Córdoba, Córdoba 14071
Email: {sventura,jrromero}@uco.es

***Abstract*—La extracción de conocimiento es un proceso complejo de varias fases. Algunas de ellas son repetitivas y consumen mucho tiempo, por lo que son susceptibles de ser automatizadas. Un ejemplo representativo es la construcción de modelos, donde la gran cantidad de algoritmos disponibles, cada uno con sus propios hiperparámetros, conforman un espacio de búsqueda de grandes dimensiones. En este contexto, surge AutoML para precisamente automatizar estas fases. La generación automática de *workflows* es una tarea de AutoML compleja que abarca la selección e hiperparametrización de los algoritmos de varias fases dando un mayor soporte el proceso de extracción de conocimiento. A diferencia de otros trabajos que fijan el orden del preprocesado, y en algunos casos el tamaño del *workflow*, nuestra propuesta genera flujos formados por un número arbitrario de algoritmos de preprocesado y uno de clasificación, permitiendo así, la generación de *workflows* más diversos. La optimización es realizada por un método de programación genética gramatical, el cual, a partir de los mejores *workflows* construye un *ensemble* y, a diferencia de otros trabajos, no solo tiene en cuenta la capacidad predictiva de los *workflows*, sino también lo diversas que son sus predicciones con respecto al resto. La propuesta ha sido empíricamente validada y comparada con TPOT y RECIPE. Los resultados experimentales demuestran que el método propuesto genera soluciones con una mayor capacidad predictiva.**

***Keywords*—AutoML, generación automática de workflows, programación genética gramatical, ensembles, clasificación**

I. Introducción

A lo largo de los años, la mayoría de empresas e instituciones han generado y almacenado un gran volumen de datos. La extracción de conocimiento útil y novedoso de dichos datos es un proceso complejo e interdisciplinar que consta de varias fases [1]. Algunas de ellas, como la interpretación de resultados o la toma de decisiones, son inherentemente humanas, ya que requieren de la experiencia, intuición y saber hacer del científico de datos. No obstante, otras son repetitivas y consumen mucho tiempo, por lo que son susceptibles de ser (al menos, parcialmente) automatizadas. Un ejemplo representativo es la construcción de modelos (p.ej. clasificadores), donde la gran cantidad de algoritmos disponibles, cada uno con su propio sesgo e hiperparámetros, forman un espacio de búsqueda de grandes dimensiones que difícilmente puede ser abordado de manera manual.

Trabajo parcialmente financiado por la Universidad de Córdoba y fondos FEDER, proyecto UCO-FEDER 18 REF.1263116 MOD, y por el Ministerio de Ciencia e Innovación, proyecto PID2020-115832GB-I00, FPU17/00799

En este contexto, se ha acuñado el término *Automated Machine Learning* [2] (AutoML) para precisamente englobar aquellos trabajos que automatizan estas fases repetitivas y costosas. Esto permite al científico de datos cubrir un abanico más amplio de posibilidades a la vez que le permite centrarse en aquellas tareas que requieren de su saber hacer e intuición. Además, en última instancia, AutoML busca acercar el proceso de extracción de conocimiento a los expertos del dominio [2]. De hecho, algunas propuestas han demostrado que AutoML puede superar a los científicos de datos en ciertas tareas como, por ejemplo, el diseño de la arquitectura de las redes neuronales artificiales [3].

AutoML es un área amplia que engloba un gran número de propuestas muy variadas, siendo la selección de algoritmos, junto con la optimización de sus hiperparámetros, dos de las tareas más comúnmente abordadas [4]. Su automatización supone una gran ayuda durante el proceso de extracción de conocimiento, aunque su uso se limita a una única fase, generalmente la construcción de modelos [5]. Cabe destacar que realizar estas tareas de manera aislada para cada fase podría perjudicar el rendimiento y dar lugar a secuencias de algoritmos inválidas. Esto se debe a la existencia de relaciones, sinergias y restricciones entre algoritmos que deben ser consideradas. En este contexto, algunos autores han propuesto métodos para automatizar la composición y optimización de secuencias de algoritmos (*workflows*), involucrando así, dos o más fases del proceso de extracción de conocimiento [6]–[8].

La composición automática de *workflows* (AWC, *Automatic Workflow Composition*) ha sido abordada fundamentalmente como un problema de optimización mediante el uso de técnicas de optimización Bayesiana y algoritmos evolutivos. En este contexto, las propuestas que emplean algoritmos evolutivos tienden a generar *workflows* más complejos y variados que aquellas que utilizan optimización Bayesiana las cuales, además, fijan la estructura del *workflow* a priori [2], [6]. No obstante, independientemente de la técnica empleada, estos trabajos restringen el tipo y orden en el que se aplican los algoritmos de preprocesado. Por otra parte, una práctica común en literatura es la de construir un *ensemble* con los mejores *workflows* generados, para lo cual se suele tener en cuenta únicamente su capacidad predictiva [2], aunque uno de los factores principales a considerar para construir buenos

ensembles es que los modelos que lo forman sean diversos [9].

En este trabajo se propone EVOFLOW, un método que aplica técnicas de computación evolutiva para automatizar la construcción de *workflows* sin asumir ningún tipo de estructura de preprocesado y buscando la generación de *ensembles* diversos, para incrementar la capacidad predictiva. Concretamente, se propone un algoritmo de programación genética gramatical [10] (G3P, *Grammar-Guided Genetic Programming*), el cual emplea una gramática libre de contexto (CFG, *Context-Free Grammar*) que define la estructura de los individuos, es decir, los *workflows*. Estos *workflows* están formados por un algoritmo de clasificación, el cual puede estar precedido por un número arbitrario de técnicas de preprocesado como, por ejemplo, selección de atributos. Cabe destacar que la gramática puede ser fácilmente extendida para dar soporte a nuevas tareas como, por ejemplo, la regresión. Durante el proceso evolutivo, EVOFLOW mantiene un archivo con los mejores *workflows* encontrados que, finalmente, son devueltos y utilizados para construir un *ensemble*. A diferencia de otras propuestas, la actualización de dicho archivo no solo considera la capacidad predictiva de los *workflows*, sino también lo diversas que son sus predicciones con respecto al resto, es decir, a cuantas muestras del conjunto de datos asigna una etiqueta distinta.

La propuesta ha sido empíricamente validada para diez conjuntos de datos de clasificación, tanto binaria como multiclase, de la literatura. Los resultados obtenidos se han comparado con los de TPOT [7] y RECIPE [11], dos herramientas para AWC que emplean programación genética y G3P, respectivamente. Los resultados experimentales demuestran como EVOFLOW supera a ambas herramientas en la mayoría de los conjuntos de datos, generando *workflows* con mejoras de la capacidad predictiva de hasta un 15%. Estos resultados son respaldados por un análisis estadístico.

El resto del trabajo se organiza como sigue. La Sección II presenta el problema de la composición automática de *workflows* y analiza el trabajo relacionado. A continuación, en la Sección III se describe el método propuesto. En la Sección IV la propuesta es validada empíricamente y comparada con otras herramientas de la literatura. Por último, las conclusiones, junto con el posible trabajo futuro, se recogen en la Sección V.

II. MARCO CONCEPTUAL Y TRABAJO RELACIONADO

La AWC implica optimizar los siguientes aspectos: (1) los algoritmos aplicados, (2) sus respectivos hiperparámetros y (3) relaciones. La selección de algoritmos (AS, *Algorithm Selection*) [12] consiste en, para un determinado conjunto de datos de entrada, predecir el algoritmo o algoritmos con mayor rendimiento. Por otra parte, la optimización de hiperparámetros (HPO, *Hyper-Parameter Optimisation*) [13] está relacionada con el ajuste de los parámetros de dichos algoritmos y que condicionan su comportamiento y rendimiento. Finalmente, las relaciones entre los algoritmos hacen referencia al orden en el que son aplicados.

Un trabajo pionero de AWC fue [6], el cual formalizó el problema combinado de AS y HPO (CASH, *Combined Algorithm Selection and Hyper-parameter optimisation*), abordándolo como un problema de optimización jerárquico, donde la selección de un determinado algoritmo (p.ej. un árbol de decisión) es un hiperparámetro que desencadena la optimización de sus correspondientes hiperparámetros (p.ej. la profundidad máxima). Este trabajo también propone Auto-WEKA, un método que utiliza técnicas de optimización Bayesiana para construir *workflows* formados por un algoritmo de selección de atributos y otro de clasificación, ambos implementados en WEKA[1]. Auto-Sklearn [2] es una propuesta similar que utiliza algoritmos de scikit-learn[2] para componer *workflows* de tres elementos. Además, utiliza meta-aprendizaje para inicializar el algoritmo de optimización Bayesiana y, finalmente, construye un *ensemble* con los mejores *workflows*. Cabe destacar que no todas las propuestas de AWC, abordan los problemas de AS y HPO de manera conjunta. Así pues, hay trabajos que, aunque aplican optimización Bayesiana para HPO, recurren a otras técnicas para realizar la AS como, por ejemplo, *Monte Carlo tree search* [14], o meta-aprendizaje y *multi-armed bandit* [15].

Como se mencionó con anterioridad, los algoritmos evolutivos son otra técnica comúnmente utilizada para la AWC. Un ejemplo representativo es TPOT [7], el cual emplea un algoritmo multi-objetivo de programación genética que busca maximizar la capacidad predictiva, a la vez que minimiza el número de algoritmos del *workflow*. A fin de acelerar el proceso de búsqueda, algunos autores han extendido TPOT evaluando primero los *workflows* en un subconjunto reducido del conjunto de datos, y usando el conjunto completo solo en aquellos *workflows* más prometedores [16]. Por otra parte, algunas propuestas han empleado gramáticas para definir la estructura de cualquier *workflow* válido, evitando así emplear tiempo en evaluar individuos inválidos. En este contexto, se han aplicado algoritmos tanto de G3P [11] como de evolución gramatical [17]. Por último y, aunque en menor medida, cabe destacar el uso de otras técnicas para la AWC como el aprendizaje por refuerzo [18], la planificación automática [8], o la inteligencia de enjambre [19].

III. MODELO DE COMPOSICIÓN AUTOMÁTICA DE WORKFLOWS

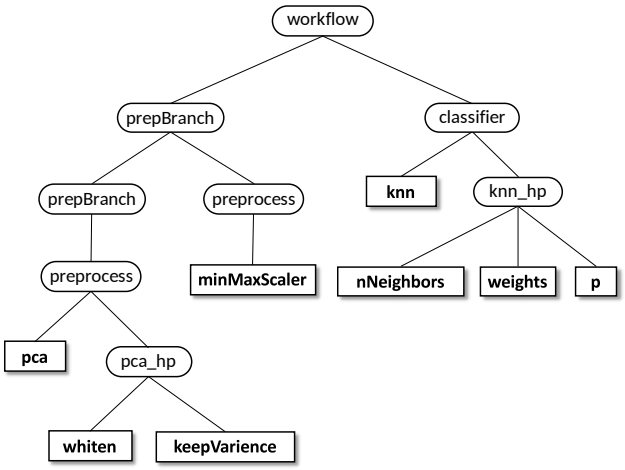
El Algoritmo 1 muestra el esquema general de EVOFLOW. Como se puede apreciar, recibe seis entradas: el número de generaciones (`maxGen`), el tamaño de la población (`popSize`), el número de *workflows* que devolverá (`arcSize`), la gramática (`cfg`), el conjunto de entrenamiento (`data`) y el tiempo máximo del que dispone el evolutivo (`timeout`). El algoritmo devuelve un conjunto (`archive`) formado por los mejores *workflows*.

Primero, se generan aleatoriamente `popSize` *workflows* en base a las restricciones impuestas por `cfg` (Sección III-A). Por otra parte, `archive` es inicializado al conjunto vacío. A continuación, se evalúan los individuos de `pop` calculando el *balanced accuracy* de cada *workflow*. Concretamente, se

[1]WEKA: www.cs.waikato.ac.nz/ml/weka/ (último acceso: 23/05/2021)
[2]scikit-learn: www.scikit-learn.org/ (último acceso: 23/05/2021)

Algoritmo 1: Esquema general de EvoFlow

```
Entrada: maxGen, popSize, arcSize, cfg, data, timeout
Salida  : archive
pop ← generateWorkflows(popSize, cfg)
archive ← ∅
evalTimeout ← timeout/10
try:
    evaluate(pop, data, evalTimeout)
    while generation < maxGen do
        pop ← select(pop ∪ extPop, popSize)
        pop ← crossover(pop)
        pop ← mutator(pop, cfg);
        evaluate(pop, data, evalTimeout)
        archive ← update(pop ∪ archive, arcSize)
        generation++
    end
    return archive
catch TimeoutException:
    return archive
end
```



(a) Genotipo

```
pca(whiten, keepVarience) → minMaxScaler() → knn(nNeighbors, weights, p)
```

(b) Fenotipo

Fig. 1. Ejemplo de un individuo

realiza una validación cruzada de cinco particiones sobre `data` para reducir el riesgo de sobreentrenamiento. Además, el tiempo de evaluación de cada individuo se limita a la décima parte de `timeout`, una práctica recurrente en la literatura [8]. A partir de este punto, el algoritmo, a lo largo de `maxGen` iteraciones, evoluciona a los individuos mediante la aplicación de operadores genéticos (Sección III-B). En primer lugar, el operador de selección escoge `popSize` individuos de la unión entre `pop` y `archive`. A continuación, el operador de cruce es aplicado, con una probabilidad definida empíricamente, sobre pares de individuos para combinar sus genotipos. Por último, se aplica el operador de mutación, también con una probabilidad definida empíricamente, cuyo objetivo es promover la diversidad entre los individuos. Tras su aplicación, `archive` es actualizado con los mejores *workflows* teniendo en cuenta, además de su capacidad predictiva, lo diversas que sean sus predicciones (Sección III-C). Cabe destacar que, si se superase el `timeout` antes de finalizar las `maxGen` iteraciones, EvoFlow finalizaría su ejecución devolviendo `archive` en su estado actual.

A. Codificación de los individuos

Cada individuo de la población se compone de dos elementos principales: (1) un genotipo, expresado en forma de árbol sintáctico, y (2) un fenotipo que representa el *workflow*. En este contexto, la Figura 1 muestra un genotipo de ejemplo, así como su correspondiente fenotipo.

Una CFG requiere la definición de conjuntos de símbolos terminales y no terminales, y las reglas de producción para derivar expresiones desde la raíz, un símbolo no terminal. En este caso, los símbolos terminales representan los algoritmos de preprocesado y clasificación, así como sus respectivos hiperparámetros, mientras que los símbolos no terminales definen los elementos derivables, incluyendo aquellos necesarios

```
<workflow>      ::= <prepBranch> <classifier> | <classifier>

<prepBranch>    ::= <preprocess> | <prepBranch> <preprocess>
<preprocess>    ::= rbfSamp <rbfSamp_hp>
                  | pca <pca_hp>
                  | minMaxScaler
                  | randomOverSampler
                  | simpleImputer
                  | ...

<rbfSamp_hp>    ::= gamma
<pca_hp>        ::= whiten keepVariance

<classifier>    ::= decTree <decTree_hp>
                  | knn <knn_hp>
                  | rndForest <rndForest_hp>
                  | bernNB <bernNB_hp>
                  | gaussianNB
                  | ...

<decTree_hp>    ::= criterion maxDepth minSamplesSplit maxFeatures
                    minSamplesLeaf classWeight
<knn_hp>        ::= nNeighbors weights p
<rndForest_hp>  ::= nEstimators criterion maxFeatures minSamplesSplit
                    minSamplesLeaf bootstrap classWeight
<bernNB_hp>     ::= alpha fitPrior
```

Fig. 2. Reglas de producción de la gramática

para construir un espacio de búsqueda jerárquico de algoritmos e hiperparámetros.

Las reglas de producción determinan los pasos de derivación que conducen a la generación de *workflows*, que finalmente se escriben en términos de símbolos terminales. La Figura 2 muestra las reglas de producción de la CFG propuesta. Nótese que, por limitación de espacio, solo se muestra un conjunto reducido, pero representativo, de reglas. La primera regla de producción determina que el símbolo raíz, *<workflow>*, puede derivarse en un clasificador, *<classifier>* que, opcional-

mente, puede estar precedido por un conjunto de algoritmos de preprocesado <*prepBranch*>. Por una parte, la rama de preprocesado se deriva en uno o varios algoritmos de preprocesado, <*preprocess*>, y sus respectivos hiperparámetros. Estos pueden cubrir varias tareas, como la extracción de atributos (p.ej. *pca*) o la normalización de los datos (p.ej. *minMaxScaler*). Nótese que no se fija ningún tamaño de *workflow* ni se establece ningún tipo de restricción sobre el orden en el que se debería realizar el preprocesado. Por otra parte, <*classifier*> se puede derivar en varios algoritmos de clasificación y sus correspondientes hiperparámetros. Téngase en cuenta que la CFG podría ser modificada, permitiendo así: (1) la generación de flujos de trabajo personalizados para, por ejemplo, usar solo modelos interpretables (p.ej. árboles de decisión); (2) el soporte de nuevas tareas de aprendizaje automático como, por ejemplo, la regresión; y (3) generar *workflows* con estructuras más complejas añadiendo, por ejemplo, varias ramas de preprocesado.

B. Operadores genéticos

El operador de selección escoge a los individuos que posteriormente se utilizarán para generar la descendencia, mediante la aplicación del cruce y la mutación. Más concretamente, se aplica un torneo binario, que toma aleatoriamente dos individuos de la unión de `pop` y `archive`, y mantiene aquel con un mayor valor de *fitness*. Este proceso se repite hasta que se seleccionen `popSize` individuos.

A continuación, el operador de cruce selecciona, aleatoriamente, un símbolo no terminal común a los dos individuos e intercambia los subárboles. Esto garantiza que los nuevos individuos sigan las restricciones impuestas por la gramática. Además, este operador controla que los individuos que se generan no excedan el número máximo de derivaciones permitido. La Figura 3 muestra un ejemplo de dicho operador, donde el símbolo no terminal común es <*prepBranch*>.

Finalmente, se aplica el operador de mutación, el cual selecciona de manera aleatoria un símbolo no terminal del árbol. A continuación, elimina el subárbol y lo reconstruye de manera aleatoria en base a la gramática evitando, de nuevo, que el número de derivaciones exceda el máximo. La Figura 4 muestra un ejemplo de aplicación de dicho operador donde el símbolo no terminal seleccionado es $< preprocess >$ por lo que únicamente se modifica el algoritmo de preprocesado.

C. Actualización del ensemble

Tras aplicar los operadores genéticos, `archive` es actualizado con los mejores, y más diversos, individuos de `pop`. Dado que en la primera generación `archive` está vacío, se añaden directamente los `arcSize` mejores individuos en base, únicamente, a su *fitness*. A continuación, para cada individuo de `archive`, se calcula lo diversas que son sus predicciones respecto al resto (Ecuación 1). Dicho valor se calcula a partir de las predicciones realizadas durante la evaluación del individuo en cada partición, las cuales son concatenadas en un único vector (x) cuya longitud es igual al número de instancias en `data`. A partir de dicho valor, se obtiene la métrica combinada de diversidad y *fitness*, que hemos denominado `divfit`, donde λ es un parámetro que controla la importancia que se le da a cada valor (Ecuación 3). A continuación, los individuos de `archive` son ordenados en base a dicha métrica.

Tabla I
CONJUNTOS DE DATOS USADOS EN LA EXPERIMENTACIÓN

	#inst.	#atrib.	#clases
Breast-Cancer (BE)	286	9	2
Car Evaluation (CAR)	1728	6	4
Caenorhabditis Elegan (CE)	478	765	2
Chen-2002 (CHEN)	179	85	2
Chowdary-2006 (CHOW)	104	182	2
Credit-G (CRED)	1000	20	2
Drosophila Melanogaster (DM)	104	182	2
DNA-No-PPI-T11 (DNA)	135	104	2
Glass (GLS)	214	9	7
Wine Quality-Red (WQR)	1599	11	10

$$div_i = \frac{1}{|pop| - 1} \sum_{\substack{j=1 \\ j \neq i}}^{|pop|} \frac{1}{|data|} \sum_{k=1}^{|data|} diff(x_{ik}, x_{jk}) \quad (1)$$

$$diff(x, y) = \begin{cases} 1 & \text{si } x = y, \\ 0 & \text{en cualquier otro caso} \end{cases} \quad (2)$$

$$divfit_i = \lambda * div_i + (1 - \lambda) * fitness_i \quad (3)$$

En las sucesivas generaciones se calcula, para cada individuo de `pop`, su diversidad frente a los de `archive` y, a partir de ella, su valor de `divfit`. A continuación, se añaden a `archive` ordenados por el valor de `divfit`, asegurando que el número de individuos no excede `arcSize`. Al realizar `maxGen` iteraciones, se devuelve `archive` formado por `arcSize` *workflows*, los cuales se utilizan para construir un *ensemble* donde se utiliza un esquema de voto mayoritario. Cabe destacar que, una vez finaliza el evolutivo, estos individuos son reentrenados utilizando el conjunto completo de entrenamiento (`data`) ya que, durante la evaluación, se realiza una validación cruzada y, por tanto, ningún *workflow* es entrenado con el conjunto completo de entrenamiento.

IV. VALIDACIÓN EMPÍRICA

EVOFLOW ha sido desarrollado en Python tomando como base el *framework* DEAP [20] (*Distributed Evolutionary Algorithms in Python*). Para los algoritmos de preprocesado y clasificación que forman los *workflows* se han tomado las implementaciones de *scikit-learn* e *imbalanced-learn*[3].

Los conjuntos de datos, así como sus particiones de entrenamiento (`data`) y prueba, usados para validar EVOFLOW son los mismos empleados por RECIPE. Cabe destacar que, para cada conjunto, existen diez particiones de entrenamiento y prueba distintas. La Tabla I muestra el número de instancias, atributos y clases de cada uno de ellos. Como se puede

[3]imbalanced-learn: *www.imbalanced-learn.org/ (último acceso: 23/05/2021)*

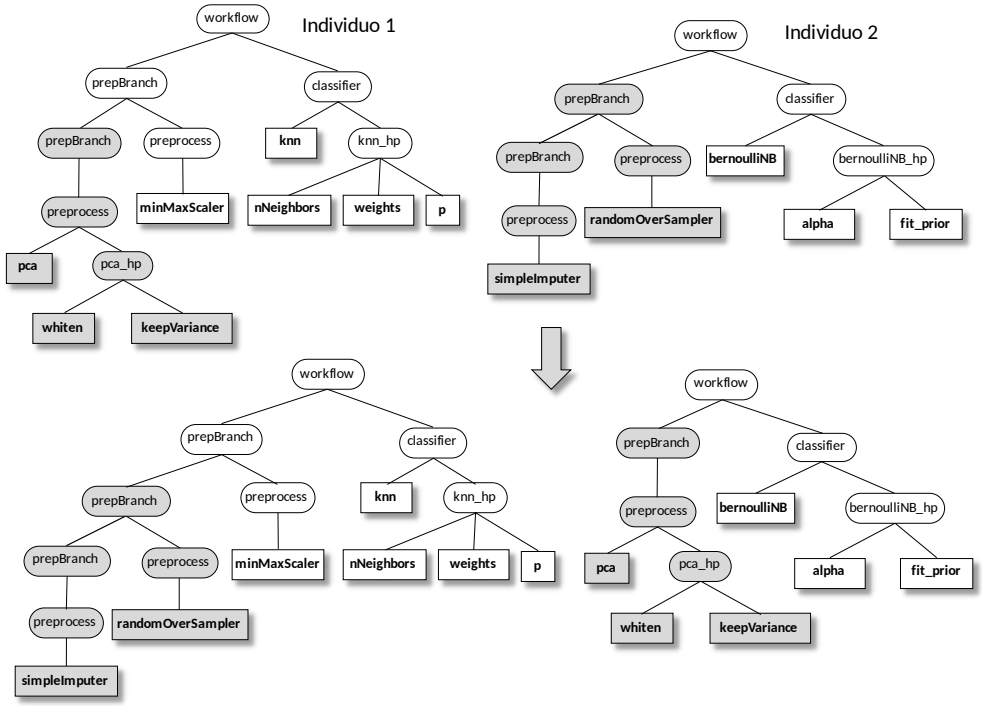


Fig. 3. Ejemplo de aplicación del operador de cruce

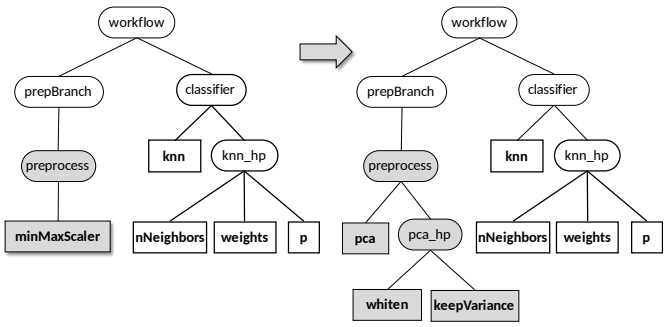


Fig. 4. Ejemplo de aplicación del operador de mutación

Tabla II
CONFIGURACIÓN EXPERIMENTAL

Parámetro	Valor
Número de generaciones	100
Tamaño de la población	100
Probabilidad de cruce y mutación	0,8
Timeout (seg)	3600
Número máximo de derivaciones	13
Número de *workflows* devueltos	20
Importancia de la diversidad (λ)	0,1

apreciar, hay problemas de clasificación binaria y multiclase. Por otra parte, la Tabla II muestra la configuración empleada por EVOFLOW, donde `timeout` se ha fijado a una hora, un valor común en la literatura [8], [11]. Es interesante destacar que el número máximo de derivaciones determina el tamaño del genotipo, es decir, del *workflow*. Este valor, junto con el número de individuos devuelto, se han fijado empíricamente en 13 y 20, respectivamente. Respecto al valor de λ, se ha fijado, también experimentalmente, con un valor de 0,1, ya que el uso de valores más altos puede llegar a provocar la generación de *ensembles* con *workflows* cuya capacidad predictiva sea demasiado baja.

Como se mencionó anteriormente, los resultados obtenidos por EVOFLOW son comparados con los de TPOT y RECIPE, las cuales utilizan programación genética y G3P, respectivamente. Cabe destacar que bajo las mismas condiciones computacionales, y sobre las mismas particiones. Además, dado que que las tres herramientas emplean algoritmos evolutivos, se han realizado para cada conjunto de datos 30 ejecuciones con diferentes semillas aleatorias. Más concretamente, para cada una de las diez particiones se han utilizado tres semillas

Tabla III
COMPARATIVA DE VALORES DE F_1 EN EVOFLOW, TPOT Y RECIPE

	EVOFLOW	TPOT	RECIPE
BE	**0,9650 ± 0,0222**	0,9552 ± 0,0281	0,9473 ± 0,0440
CAR	0,9959 ± 0,0122	**0,9964 ± 0,0114**	0,9405 ± 0,0694
CE	**0,5948 ± 0,0535**	0,5708 ± 0,0642	0,5663 ± 0,0580
CHEN	**0,9594 ± 0,0450**	0,9328 ± 0,0569	0,9336 ± 0,0523
CHOW	**1,0000 ± 0,0000**	0,9825 ± 0,0399	0,9933 ± 0,0365
CRED	**0,6804 ± 0,0489**	0,6732 ± 0,0471	0,5913 ± 0,0771
DM	**0,5722 ± 0,1118**	0,5207 ± 0,1229	0,5403 ± 0,1165
DNA	**0,5669 ± 0,1228**	0,5117 ± 0,1419	0,5436 ± 0,1446
GLS	**0,7208 ± 0,1199**	0,6616 ± 0,1227	0,6648 ± 0,1220
WQR	0,3294 ± 0,0493	0,3291 ± 0,0548	**0,3356 ± 0,0265**

distintas.

La Tabla III recoge, para cada conjunto de datos y herramienta, la media y la desviación estándar de la medida F_1, la cual es la media armónica de la precisión y la exhaustividad de un clasificador. Como puede apreciarse, EVOFLOW alcanza valores superiores en ocho de los diez conjuntos de datos, mientras que TPOT y RECIPE solo obtienen los mejores resultados en un conjunto cada uno (CAR y WQR, respectivamente). Además, en estos casos la diferencia de F_1 respecto a EVOFLOW es inferior a 0,01. Sin considerar a EVOFLOW, RECIPE ganaría en seis conjuntos. Por otra parte, las mayores diferencias a favor de EVOFLOW, se dan para CRED y DM, donde el incremento en rendimiento es del 15 y 9%, respectivamente. Respecto a la de desviación estándar, las tres herramientas obtienen valores muy similares en la mayoría de los conjuntos de datos, siendo EVOFLOW el que tiende a obtener el valor más bajo. Por lo general, dichos valores son reducidos, demostrando que las tres herramientas son bastante robustas a la semilla aleatoria.

Por último, se ha aplicado un test de Friedman para comprobar si existen diferencias significativas entre los tres métodos. Con un *p-value* de 0,0074, podemos rechazar la hipótesis nula y realizar un análisis post-hoc. En concreto, hemos aplicado el test de Hochberg obteniendo unos *p-values* ajustados de 0,0073 y 0,0139 para TPOT y RECIPE, respectivamente. Dado que ambos valores son inferiores a 0,05, podemos concluir que EVOFLOW obtiene resultados significativamente superiores a los obtenidos por TPOT y RECIPE.

V. CONCLUSIONES Y TRABAJO FUTURO

En este trabajo se propone EVOFLOW, un método para la composición automática de *workflows* mediante el uso de algoritmos evolutivos. Concretamente, se propone un algoritmo de programación genética gramatical, el cual emplea una gramática para definir la estructura de cualquier *workflow* válido. A diferencia de otros trabajos, nuestra propuesta permite cualquier combinación de algoritmos de preprocesado. Además, construye un *ensemble* con los mejores *workflows* encontrados basándose, además de en su capacidad predictiva, en lo diversas que son sus predicciones.

La propuesta ha sido empíricamente validada para diez conjuntos de datos. Los resultados se han comparado con los de TPOT y RECIPE, dos propuestas evolutivas reconocidas para la composición automática de *workflows*, mostrando que EVOFLOW es significativamente superior en base a la medida de rendimiento predictivo F_1.

Como trabajo futuro se planea incorporar nuevos operadores genéticos específicos del problema de la composición de *workflows*. Asimismo, sería interesante añadir nuevos algoritmos a la gramática que permitan dar soporte a otras tareas de aprendizaje automático. También consideramos de interés el desarrollo de propuestas interactivas que incorporen la experiencia y saber-hacer del científico de datos al proceso de optimización. Por último, podría interesar integrar EVOFLOW con herramientas de amplio uso como KNIME o RapidMiner como forma de acercar el método al experto del dominio.

REFERENCIAS

[1] U. M. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, "From data mining to knowledge discovery: An overview," in *Advances in Knowledge Discovery and Data Mining*, U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, Eds. Menlo Park, CA, USA: American Association for Artificial Intelligence, 1996, pp. 1–34.

[2] M. Feurer, A. Klein, K. Eggensperger, J. Springenberg, M. Blum, and F. Hutter, "Efficient and robust automated machine learning," in *Advances in neural information processing systems*, 2015, pp. 2962–2970.

[3] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le, "Learning transferable architectures for scalable image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8697–8710.

[4] F. Hutter, L. Kotthoff, and J. Vanschoren, *Automated machine learning: methods, systems, challenges*. Springer Nature, 2019.

[5] B. Bilalli, A. Abelló, T. Aluja-Banet, and R. Wrembel, "Automated data pre-processing via meta-learning," in *International Conference on Model and Data Engineering*, 2016, pp. 194–208.

[6] C. Thornton, F. Hutter, H. H. Hoos, and K. Leyton-Brown, "Auto-weka: Combined selection and hyperparameter optimization of classification algorithms," in *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2013, pp. 847–855.

[7] R. S. Olson, N. Bartley, R. J. Urbanowicz, and J. H. Moore, "Evaluation of a tree-based pipeline optimization tool for automating data science," in *Proceedings of the Genetic and Evolutionary Computation Conference 2016*, 2016, pp. 485–492.

[8] F. Mohr, M. Wever, and E. Hüllermeier, "Ml-plan: Automated machine learning via hierarchical planning," *Machine Learning*, vol. 107, no. 8-10, pp. 1495–1515, 2018.

[9] Y. Bi, "The impact of diversity on the accuracy of evidential classifier ensembles," *International Journal of Approximate Reasoning*, vol. 53, no. 4, pp. 584–607, 2012.

[10] R. I. Mckay, N. X. Hoai, P. A. Whigham, Y. Shan, and M. O'Neill, "Grammar-based genetic programming: A survey," *Genetic Programming and Evolvable Machines*, vol. 11, no. 3-4, pp. 365–396, 2010.

[11] A. G. de Sá, W. J. G. Pinto, L. O. V. Oliveira, and G. L. Pappa, "Recipe: a grammar-based framework for automatically evolving classification pipelines," in *European Conference on Genetic Programming*. Springer, 2017, pp. 246–261.

[12] J. R. Rice, "The algorithm selection problem," in *Advances in computers*. Elsevier, 1976, vol. 15, pp. 65–118.

[13] L. Yang and A. Shami, "On hyperparameter optimization of machine learning algorithms: Theory and practice," *Neurocomputing*, vol. 415, pp. 295–316, 2020.

[14] H. Rakotoarison, M. Schoenauer, and M. Sebag, "Automated machine learning with monte-carlo tree search," *arXiv preprint arXiv:1906.00170*, 2019.

[15] Z. Shang, E. Zgraggen, B. Buratti, F. Kossmann, P. Eichmann, Y. Chung, C. Binnig, E. Upfal, and T. Kraska, "Democratizing data science through interactive curation of ml pipelines," in *Proceedings of the 2019 International Conference on Management of Data*, 2019, pp. 1171–1188.

[16] P. Gijsbers, J. Vanschoren, and R. S. Olson, "Layered tpot: Speeding up tree-based pipeline optimization," in *2017 International Workshop on Automatic Selection, Configuration and Composition of Machine Learning Algorithms*, 2017, pp. 49–68.

[17] S. Estévez-Velarde, Y. Gutiérrez, Y. Almeida-Cruz, and A. Montoyo, "General-purpose hierarchical optimisation of machine learning pipelines with grammatical evolution," *Information Sciences*, vol. 543, pp. 58–71, 2020.

[18] A. Elkholy, F. Yang, and S. Gustafson, "Interpretable automated machine learning in maana™ knowledge platform," in *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*, 2019, p. 1937–1939.

[19] A. Díaz-Pacheco and C. A. Reyes-García, "Full model selection in huge datasets and for proxy models construction," in *Mexican International Conference on Artificial Intelligence*. Springer, 2018, pp. 171–182.

[20] F.-A. Fortin, F.-M. De Rainville, M.-A. G. Gardner, M. Parizeau, and C. Gagné, "Deap: Evolutionary algorithms made easy," *The Journal of Machine Learning Research*, vol. 13, no. 1, pp. 2171–2175, 2012.

Autonomous Learning: Tackling Expansive Mutating Domains

Harshal V Bharatia
ACM Student Member,
Plano, Texas, USA
Email: hbharatia.edu@gmail.com

***Abstract*—This paper presents an autonomous approach to knowledge refinement wherein a system automatically determines what can be improved and trains itself without explicit guidance to do so. It enables solutions with smartness that goes beyond initial training, when domain is expansive and high quality training may not be feasible for entire expanse of the domain, or when spontaneous mutations alter various aspects of the domain. A novel approach with automatically built dynamic ensembles of models target improvisations, such as improvements based on past knowledge or trends reflected in the results, and overshadows sub-optimal performance of existing model in a portion of input feature space. A model selector automatically identifies an optimal model for a query using reinforcement learning. This alleviates need for complex monolithic models requiring extensive training, automatically adapt to changes in the domain without destabilizing the models, offer better control over query performance, and can guide exploration when exploitation becomes inadequate. Experimental results show that performance possible with this approach exceeds that of a standalone deep-learning network, both with minimal training in early stage and with experience, adapting quickly to mutations while providing superior performance.**

I. Introduction

Today supervised deep learning models often come up short in many real-world applications, such as when predictions are to be performed for inputs with inadequate training, when the problem domain undergoes changes or model was not trained for a previously unknown trend in the results. Moreover, when the problem domain is expansive or incurs frequent and many times spontaneous changes, it leads to sub-optimal performance as high quality training may not always be readily available for the entire expanse of the problem domain. As the domain becomes expansive, the model complexity also increases, making it difficult to retain acceptable performance over the entire problem domain. Supervised training solutions cannot automatically adapt to domain changes that were not effectively captured in the training data. However, many pragmatic real-world solutions need to adapt to changes observed in the domain. Mutations in the problem domain requires retraining and when such mutations can occur at any time, such retraining becomes quite tricky as it is not always possible to judge the time and scope of these changes.

A novel method of using one or more base models overshadowed by autonomously built targeted models is demonstrated here to effect automatic model refinement that addresses issues such as the ones discussed above for expansive and mutating domains. A system automatically creates dynamic ensembles comprising automatically trained weak-models and each dynamic ensemble targets a subset or a specific aspect of the problem domain. A model selector uses reinforcement learning for optimal selection of a model from available models and in doing so, effectively overrides non-performant aspects of the base model. With reinforcement learning, when environment changes, the optimal input-output pairs learned earlier for model selection becomes invalid but new behaviors are strengthened or weakened using a result-oriented reward strategy.

II. Related Work

[1], [2] present training approaches for large model but do not address lack of high quality training data, pragmatic retraining, localize effects of mutations and leveraging preexisting knowledge on-demand. They do not address automatic learning beyond the initial training. Ensemble techniques in [3], [4], [5], and [6] also do not address these issues for expansive and changing domain.

[7] presents dynamic ensemble learning using automatically generated weak-models. However, no single criterion is sufficient to effectively evaluate competency of a weak-model over entire problem range and [8] addresses it and further describes a dynamic ensemble using meta-learning. Single ensemble performance improves, but changes affect entire ensemble and fail to address expansive domains.

[9] introduces many different techniques for reinforcement learning using dynamic programming, monte carlo and temporal different learning. However, these techniques cannot directly address the problem facing expansive problem domains which are otherwise amenable to supervised learning, and therefore do not address the technique such as the novel use of temporal difference learning for model selection proposed in this paper for optimal decision-making.

III. Proposed Approach

One or more state-of-art machine learning base models represent the problem domain. The large input feature space of the expansive problem domain is effectively segmented by dynamically and automatically created small better performing models that overshadow weak portions of base model. Each small model excels as it targets only a small slice of the huge input feature space. As these small models may target

overlapping areas of the input feature space, a domain-specific reward driven strategy is used to learn how to select an appropriate model for a query prediction using reinforcement learning (RL).

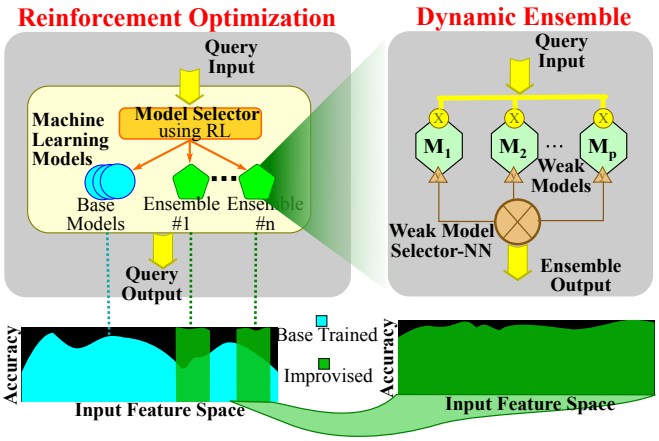


Fig. 1. Autonomous Learning Model

A result-driven relevancy analysis constantly looks for exceptional results, leverages past knowledge, identifies new trends, and detects mutations in problem domain - thereby autonomously driving creation of new better performing targeted models. As shown in Figure 1, targeted dynamic Ensemble #n, is the small model overshadowing a bad performing slice of the base model input feature space. The model selector selects a model and hence dynamically controls how the input feature space of the problem domain gets segmented without segmenting any models. The collection of base models, dynamically created dynamic ensembles each targeting small subsets of the input feature space, and the model selector together serve as the effective autonomous learning model.

When the input feature space of many models overlap, it is essential to determine which model can optimize prediction for a query. Each dynamic-ensemble may use different input features, and hence an optimal model for the prediction cannot be trivially determined. Moreover, as problem domain changes, a model's relevance for such query prediction also changes. Therefore, a novel reinforcement learning (RL) based model selector is used to learn how to select an optimal model for a query. As new dynamic ensembles can be automatically created, the model selector uses a novel way to incorporate the new ensemble for future predictions.

With continuous automatic refinements, this method of autonomous learning enables completely overlapping the non-performant portions of a base model with one or more highly performant models that target tiny slices of the input feature space. The RL model selector allows this overlap to be completely dynamic and can effectively reinforce a non-performant portion of base model using more than one tiny models as reinforced by rewards from the past results. Furthermore, additional tiny models may be lazily added as needed, making this technique highly versatile for fast changing domains.

A. Dynamic Ensemble

Relevancy Analysis obtains most relevant data from entire dataset using principal input features periodically or on-demand. Without loss of generality, such analysis may be performed using various means such as data analytics, and domain specific criteria. On sufficient availability of relevant *improvisation training data* $\mathbb{T}$, a new ensemble gets generated.

In order to build the ensemble, identify weak-models $m_1, \ldots, m_n$ to be created based on size of $\mathbb{T}$. Select a principal factor q as identified by relevancy analysis, divide its range of values into p regions of competence, and create a region of competence $R_i = [q_{L_i}, q_{H_i}]$, $1 \leq i \leq p$ using probability mass function (PMF) of q. Divide improvise data set $\mathbb{T}$ into p data sets $\mathbb{T}_i, 1 \leq i \leq p$ such that each $X_j \in \mathbb{T}_i$ has $q_{L_i} \leq q(X_j) < q_{H_i}$, redistributing samples if needed for even distribution. Bootstrap m_i with $\mathbb{T}_i$ for each $i, 0 \leq i \leq n$. Process similarly if there are multiple principal factors.

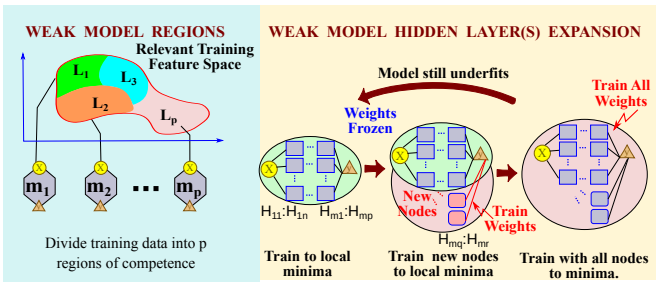


Fig. 2. Weak-Model Generation

1) Weak-Model Generation: For each model m_i first determine number of input and output nodes using query input and output. As shown in Figure 2, create a single hidden layer comprising minimum number of hidden nodes h_{min}. Train model m_i using $\mathbb{T}_i$. Model is under-fitting if test and train accuracy decreases but remain below acceptable threshold. While m_i under-fits, use Sequential Network Construction to expand hidden layer(s) until m_i reaches acceptable accuracy.

Sequential network construction adds c nodes to last hidden layer H_m. If $|H_m| + c \geq h_{max}$, where h_{max} is maximum hidden layer size, create a new hidden layer with h_{min} nodes. *Freeze* weights of existing nodes and train model m_i using $\mathbb{T}_i$ so that the new c nodes reach their local minima. This reuses past training of existing nodes and expedites model expansion. Finally, *unfreeze* entire network and retrain it using $\mathbb{T}_i$. This allows exploring all minimas globally.

By iteratively adding more nodes until model no longer under-fits, above algorithm ensures that the weak-model has a low bias with acceptable accuracy while avoiding over-fitting and ensuring that the variance is not very high [10]. When principal factor based segmentation of improvise data set $\mathbb{T}$ is not possible, a collection of weak-models can be built with dataset $\mathbb{T}$ using negative correlation learning as described in [11].

2) Ensemble Weak-Model Selector Generation: Weak-model selector λ in Figure 3 selects how weak-models con-

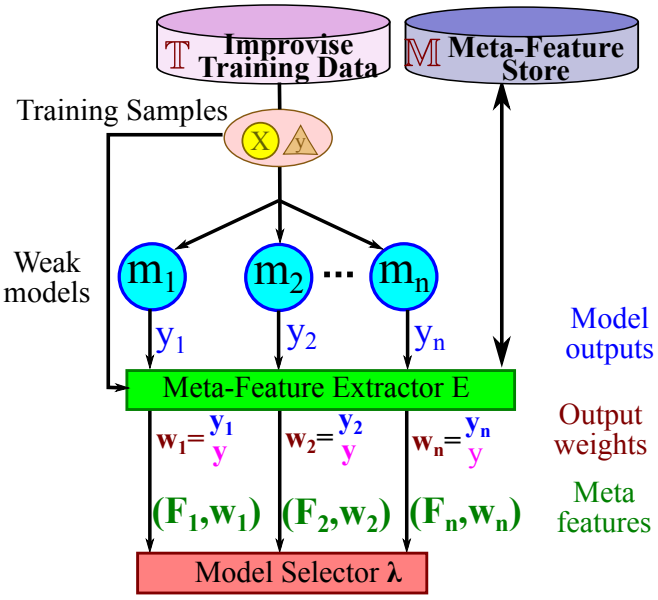


Fig. 3. Model Selector Generation

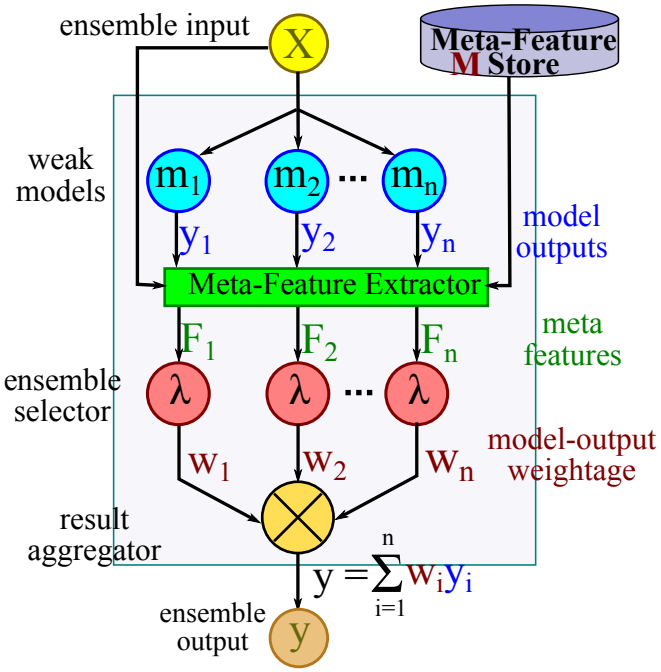


Fig. 4. Ensemble Predictor

tribute to ensemble output. It is a fully connected neural network that takes a weak-model and meta-feature set for this weak-model to predict weightage of weak-model for ensemble output.

Algorithm 1 Selector Generation Algorithm

1: **for** each weak-model $m_i, 1 \leq i \leq n$ **do**
2: **for** each training sample $\{X_j, y_j\} \in \mathbb{T}_i$ **do**
3: Compute output y_i for X_j using model m_i
4: Extract meta-feature set F_i for X_j and y_i
5: Compute output weightage as $w_i = y_i/y_j$
6: Train selector λ with sample $\{F_i, m_i\} \rightarrow w_i$
7: **end for**
8: **end for**

3) Ensemble Predictor: The selector in Figure 4 helps combine outputs of weak-models using meta-features for the guidance query.

Algorithm 2 Ensemble Predictor Algorithm

Input : Guidance query X_j
Output: Ensemble output y

1: **for** each weak-model $m_i, 1 \leq i \leq n$ **do**
2: Compute output y_i for input X_j using model m_i
3: Extract meta-feature set F_i for X_j and y_i
4: Using selector λ, obtain w_i: $\{F_i, m_i\} \rightarrow w_i$
5: **end for**
6: Aggregate result using y_i and w_i for each m_i:
- For regression: $y = \sum_{i=1}^{n} w_i y_i$
- For classification:
$\hat{y} = \arg\max_j \sum_{i=1}^{n} w_i y_{ij}$
$y_i = \{y_{i1}, y_{i2}, ..., y_{ic}\}$, and $\hat{y}$: overall probability of c classes.

4) Evaluating Weak-Model Competence: Meta-feature is a criterion to estimate competency of a weak-model to produce correct output for an input sample. A competent weak-model's output must have bigger effect on ensemble's output. As shown in [12], [8] no single criterion estimates the most competent weak-model and so multiple criteria are needed to overcome weaknesses of one or more criterion. These criteria are based on following techniques for regression and classification weak-models.

Region of Competence: Weak-models tend to perform same for similar inputs. Small region in feature space around input sample X_j is called region of competence R_j. It can be computed using K-NN or clustering techniques. If weak-model was accurate for k_N similar inputs as X_j, it is competent to evaluate X_j. Algorithms include Overall Local Accuracy (OLA), Local Classifier Accuracy (LCA), Modified Local Accuracy, K-Nearest Oracles similar to ones in [8]. Meta-features include local model accuracy f_1, local model consensus f_2, overall local accuracy f_3 and overall local confidence f_4. However, its effectiveness is limited by performance of algorithm defining region of competence.

Decision Templates: For competent weak-model, similar inputs tend to produce similar output. For an input sample X_j, output profile O_j is created using o_{ji} outputs of each weak-model $m_i \in M$, $1 \leq i \leq n$. k_p input samples with nearest output profiles to O_j are found. Here, only weak-models that have accuracy above certain threshold for the k_p samples are considered competent. Algorithms include K-Nearest Output Profile and Multiple Classifier Behaviour (MCB) similar to ones in [8]. Meta-features include decision accuracy f_5, decision confidence f_6. However, these techniques consider

only global performance of weak-models ignoring any local competency.

Extent of Consensus: Weak models that produce same output tend to be competent. Meta-features include model consensus f_7. However, a tie ends up being randomly resolved and pool of ensembles greatly increase computational complexity.

B. Model Selector

A strategy-based model selector is used to determine best model to provide guidance. The input feature-space of one or more models can overlap, and hence may be relevant for a query. Due to large number of input parameters, selecting model for a query is complex and not deterministic, and hence requires learning. The selector starts using a principal-feature based segmentation to select models until optimal reinforcement learning (RL) based selection is ready. Unlike [13], [14], RL is used to select models, thereby achieving improvements without disrupting the model's characteristics.

The RL based model selection is based on a Markov Decision Process (MDP) with objective of maximizing expected return comprising a set of states S, set of possible actions A, scalar reward $R_a(s, s')$ received after transition from state $s \in S$ to $s' \in S$ due to action $a \in A$. An episode comprises a sequence of guidance queries for an aspect of the problem domain. A query's input parameters determine an MDP state for the domain-specific aspect and an action is taken to select a learning model that provides prediction for the query. This prediction provided for the query effects the domain environment and its consequences form the basis of computing the reward for this prediction. The effect on the domain environment may affect the next query in the episode, causing transition to next MDP state for the domain-specific aspect. The goal is to maximize expected return for the entire episode.

The states are based on automatically segmenting input feature space $\mathbb{F}_B$ of the problem domain F_B. Consider input feature $f_i \in F_B, 1 \leq i \leq n_F$. Values along one or more input feature in $D_B \subset F_B$ get segmented and an RL state $s_{Bi} \in S_B, 1 \leq i \leq n_{Bs}$ is defined for input-feature space $\mathbb{F}_B$. When new ensemble gets created, additional states $s_{Ci} \in S_C, 1 \leq i \leq n_{Cs}$ representing the ensemble's feature space $\mathbb{F}_C$ can be lazily added by segmenting along one or more input features in $D_C \supset D_B$. In these new states, the additional model is available in addition to the models that existed before the addition of the new ensemble.

An MDP action $a \in A$ represents selecting a model available to handle guidance request. Available models include a base model and any dynamic ensembles that are created automatically. The selector learns using a feedback corresponding to a query input and its result based on a domain-specific criteria, thereby reinforcing its choice for the selected action in an MDP state. Temporal difference double-Q reinforcement learning is used to reduce maximization bias as in [9]. Thus model selector learns to select optimal action in any state resulting in query processing using optimal model. A Deep Q-Network (DQN) may instead be used if number of states become large as in [15] and [16].

IV. Results and Discussion

Autonomous learning approach was tested with diverse expansive mutating domains including a sports coach-sidekick to maximize score, medical assistant for Covid-19 treatment that maximizes recovery probability in shortest time, and for Anterior Cruciate Ligament (ACL) surgery to help make decisions that provides quickest recovery. Autonomous learning exhibited similar superior efficacy improvements for these test domains over state-of-art deep learning models, further affirming the efficacy of autonomous learning proposed in this paper. A detailed result analysis is presented here for a sample problem domain of an American Football coach-sidekick that predicts play-calls to maximize points for an offensive drive to demonstrate the effectiveness of the approach proposed in this paper.

A. Experimental Setup

The game for the sample problem domain of American Football involves many strategies based on player capabilities and constrained by numerous intricate rules, making it an expansive domain with spontaneous mutations due to change in playing conditions or new rules. The testing is based on play-by-play dataset from nflscrapR [17] for play-calls and game state and player data from player-stats dataset [18].

The input-feature set comprises of game-state and numerous player capabilities based on play positions and player statistics. Output is a play-call. Game-state includes current score, time left, yard-line, yards to go and timeouts. An episode comprises of plays for offense team starting with ball possession until they score points or loose ball possession. The efficacy of a model μ for a specific play-call prediction is evaluated based on number of remaining plays n_p in episode and the points scored s_o at end of the episode. $\mu = w_s \frac{s_o - max(0, s_o - 7)}{7} \times w_p \frac{max(k_l, k_u - max(k_l, n_p))}{k_u - k_l}$, where k_u is upper limit to 50% rewards and k_l is lower limit of 100% rewards based on n_p and defaults to $k_u = 30$, $k_l = 10$, $w_s = 1$ and $w_p = 1$. A game-state neural network is used to predict effect of a play-call: the time-left, scored points and yards to go for the input-features. It is very extensively trained and allows realistic simulation of game-state transitions for evaluating efficacy of a play-call in a simulated episode.

A domain-specific reward function trains the RL based model selection. It attempts to maximize the score by rewarding an offense for actions that improves probability of scoring points at end of an episode. It rewards gained yards y_g, number downs n_d, and scored points s_o but penalizes a turnover t, opponent points s_d and punt return yards y_p. Thus $R = w_{yg} y_g + (c_{nd} - w_{nd} n_d) + w_{so} s_o - (w_t t + w_{sd} s_d + w_{yp} y_p)$. Each factor is weighted with default value of $w_{yg} = 0.25$, $c_{nd} = 6$, $w_{nd} = 2$, $w_{so} = 4$, $w_t = 16$, $w_{yp} = 4$

Autonomous learning was tested against different manually optimized state-of-art supervised learning baseline models

such as deep learning (DL) neural network and gradient-boosted decision tree. The goal is to compare play-call efficacy while maintaining comparable model size and structure. Consider an autonomous base model with l_a fully connected layers each with n_{al} nodes, e_a dynamic ensembles each having w_e weak models on average with n_{we} nodes. If baseline DL model is structured around l_a layers, layer size is constrained to $\frac{l_a n_{al}+e_a(w_e n_{we})}{l_a}$ nodes on average. If a gradient boosted decision tree has $e_a + 1$ trees each is constrained around $\frac{l_a n_{al}+e_a(w_e n_{we})}{e_a+1}$ nodes. Test data comprised play-by-play nflscrapR dataset for 2017 and 2018 seasons with 70K/20K train/validate split and rest of years 360K samples trained game-state neural network. Autonomous learning trained with 20K initial samples followed by refinements with 50K samples. Baseline models used 20K initial samples followed by five retraining session 10K each. Similar results were obtained with dataset for other seasons.

To create the dynamic ensembles, training samples that resulted in predictions with outstanding efficacy were identified as an optimal training sample subset. A training spread represents a percentage of standard deviation of the principal factor for choosing training data samples around a member of the optimal training sample subset. Training samples with principal factor bounded within the training spread were identified and repeatedly used until system created ensembles based on the results of these samples. The efficacy improvement was measured using validation dataset as percentage improvement in efficacy for an episode over baseline model.

B. Comparisons and Analysis

Table I summarizes performance improvements. In Figure 5 the change-units represent ensemble, episode-side, or mutations depending on the test. The efficacy improvement was recorded as average efficacy improvement of a play-call with autonomous learning over deep learning model. As number of targeted ensembles increase, the accuracy increases as ensembles overshadow sub-optimal model aspects, even after retraining the deep model. Increase of ensembles included result-driven new ensembles and ensembles based on previous games, other leagues and results from other game-strategy analytics. Longer episodes benefit more due to optimal ensemble selection using reinforcement learning. Specialized mutation ensembles focus entirely on explicit aspects of mutation whereas full retraining tend to average out new changes irrespective of number of mutations. Retraining loses pre-mutation state and hence perform worse for those queries.

Figure 6 shows effect of number of dynamic ensembles and spread of principal input factor in the training dataset used to train the ensembles. Optimal performance is with 20 - 25 ensembles and 20 - 30% spread shown in orange. Narrow spread exhibit low efficacy as although model is effective, not many queries use them, thereby limiting its effect on efficacy. Model effectiveness diminishes as spread increases and therefore, it has less impact on improving efficacy. As number of dynamic ensembles increases, model overlap increases and therefore, overall performance depends more on ability of the model selector to select optimal model. With sufficient dynamic ensembles having moderate spread, there is a significant efficacy improvement. This result is very significant as it demonstrates that automatically creating dynamic ensembles using a result-driven approach enabled the system to refine its knowledge quite significantly beyond the

TABLE I
PERFORMANCE TEST SUMMARY

Purpose	*Baseline*	*Type*	*Values*
Number Ensembles	Retrained DL	Efficacy Improve	Mean:17.01% σ:5.09%
Episode Accuracy	Retrained DL	Efficacy Improve	Mean:10.26% σ:2.95%
New Mutations	Retrained DL	Efficacy Improve	Mean:10.85% σ:1.57%
Post-mutation unmutated	Retrained DL	Efficacy Improve	Mean:10.04% σ:1.09%
Reinforce New Ensemble	DL Base, ε=0.1	Accuracy	91.32-99.18%, Attempts=4300

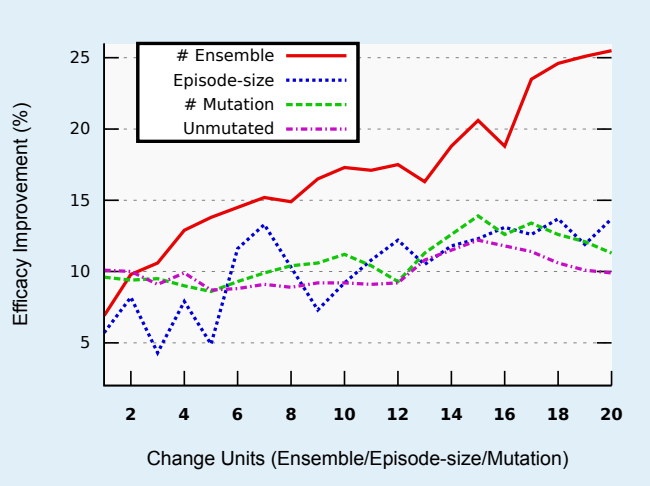


Fig. 5. Efficacy Improvements

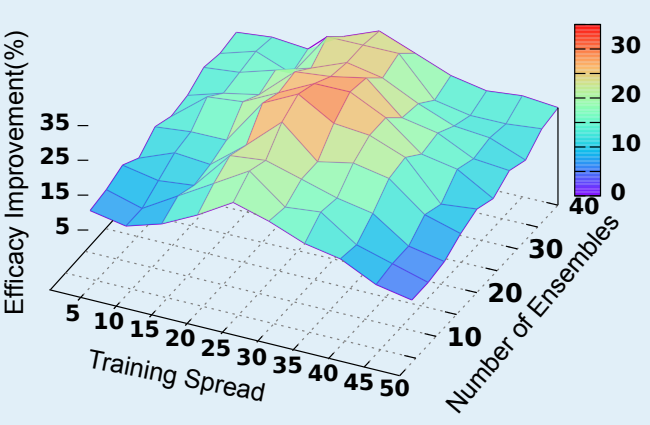


Fig. 6. Dynamic Ensemble Effects

initial training and such refinement led to better performance as more dynamic ensembles gets created. Moreover, training the dynamic ensemble using a dataset spanning a tiny fraction of expansive input feature space is effective. Hence it can effectively train for specific targeted purposes such as a specific trend in current results, leverage optimizations from other knowledge sources, and quickly adapt to mutations.

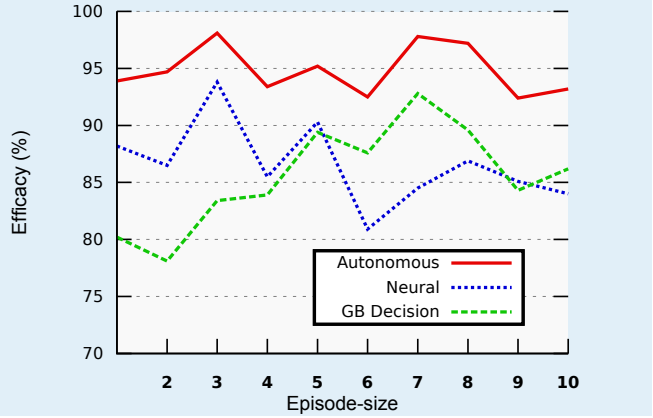


Fig. 7. Efficacy Comparison by Episode Size

Figure 7 shows superior efficacy of autonomous learning against baseline neural network and gradient-boosted decision tree. Different episodes of varying sizes ending with scored points were picked and their start game-state were used for this test. The average efficacy of first play-calls in episodes picked for same size in used to compare how play-call efficacy changes by episode size for each model. Autonomous learning offered play-calls with more than 92% efficacy and outperformed baseline models in excess of 10% demonstrating that targeted ensembles resulted in predicting fewer more effective play-calls and episodes with better scores. Continuous autonomous refinements, led to effective targeted dynamic ensembles that significantly increases its efficacy.

V. Conclusion

With the novel autonomous learning approach demonstrated in this paper, it becomes possible for a system to provide guidance that exceeds the training of its base models. Using a result-driven approach, the system continuously and automatically identifies portion of input feature space for improvement, builds dynamic ensemble targeting this portion and using reinforcement learning learns to use a model for optimizing a prediction. Furthermore, it becomes possible to selectively tap into other knowledge sources for training the targeted dynamic ensembles thereby overcoming need for extensive and full initial training, something not always possible for expansive domains. This approach excels at adapting to new trends and changes in the problem domain as continuous automatic improvements selectively eclipses non-performant outdated models. Result-driven targeted ensembles improved accuracy significantly compared to full model retraining. Adapting to changes in certain aspects of problem domain is immediate and with significant improvement over full model retraining. Reinforcement learning based model selector enables using optimal model and supports exploration based on alternate models when exploitation is inadequate. Ability to switch models using RL retains model's characteristics without disrupting use of state-of-art techniques to manage the model. With autonomous refinement, it becomes possible to have effective and pragmatic learning models for many real-world applications, and makes it feasible to deal with the expansive mutating problem domains.

References

[1] J. Dean, G. Corrado, R. Monga, K. Chen, M. Devin, M. Mao, M. Ranzato, A. Senior, P. Tucker, and K. Yang, "Large scale distributed deep networks," *Advances in neural information processing systems*, vol. 25, pp. 1223–1231, 2012.

[2] P. Goyal, P. Dollár, R. Girshick, P. Noordhuis, L. Wesolowski, A. Kyrola, A. Tulloch, Y. Jia, and K. He, "Accurate, large minibatch sgd: Training imagenet in 1 hour," 2018.

[3] H. Chen and X. Yao, "Regularized negative correlation learning for neural network ensembles," *IEEE Transactions on Neural Networks*, vol. 20, no. 12, pp. 1962–1979, 2009.

[4] Y. Liu and X. Yao, "Ensemble learning via negative correlation," *Neural networks*, vol. 12, no. 10, pp. 1399–1404, 1999.

[5] M. Shahjahan and K. Murase, "A pruning algorithm for training cooperative neural network ensembles," *IEICE TRANSACTIONS on Information and Systems*, vol. 89, no. 3, pp. 1257–1269, 2006.

[6] M. Alhamdoosh and D. Wang, "Fast decorrelated neural network ensembles with random weights," *Information Sciences*, vol. 264, pp. 104–117, 2014.

[7] K. M. R. Alam, N. Siddique, and H. Adeli, "A dynamic ensemble learning algorithm for neural networks," *Neural Computing and Applications*, vol. 32, no. 12, pp. 8675–8690, Jun. 2020. [Online]. Available: http://link.springer.com/10.1007/s00521-019-04359-7

[8] R. M. O. Cruz, R. Sabourin, and G. D. C. Cavalcanti, "META-DES.H: A Dynamic Ensemble Selection technique using meta-learning and a dynamic weighting approach," in *2015 International Joint Conference on Neural Networks (IJCNN)*. Killarney, Ireland: IEEE, Jul. 2015, pp. 1–8. [Online]. Available: http://ieeexplore.ieee.org/document/7280594/

[9] R. S. Sutton and A. G. Barto, *Reinforcement learning: an introduction*, second edition ed., ser. Adaptive computation and machine learning series. Cambridge, Massachusetts: The MIT Press, 2018.

[10] T. Ash, "Dynamic node creation in backpropagation networks," *Connection science*, vol. 1, no. 4, pp. 365–375, 1989.

[11] T. Higuchi, Xin Yao, and Yong Liu, "Evolutionary ensembles with negative correlation learning," *IEEE Transactions on Evolutionary Computation*, vol. 4, no. 4, pp. 380–387, Nov. 2000. [Online]. Available: http://ieeexplore.ieee.org/document/887237/

[12] R. M. Cruz, R. Sabourin, and G. D. Cavalcanti, "Meta-des.oracle: Meta-learning and feature selection for dynamic ensemble selection," *Information Fusion*, vol. 38, pp. 84 – 103, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S156625351730101X

[13] J. D. Williams, K. Asadi, and G. Zweig, "Hybrid code networks: practical and efficient end-to-end dialog control with supervised and reinforcement learning," *arXiv preprint arXiv:1702.03274*, 2017.

[14] R. Calvo and M. Figueiredo, "Reinforcement learning for hierarchical and modular neural network in autonomous robot navigation," in *Proceedings of the International Joint Conference on Neural Networks, 2003.*, vol. 2. IEEE, 2003, pp. 1340–1345.

[15] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing atari with deep reinforcement learning," *arXiv preprint arXiv:1312.5602*, 2013.

[16] H. Van Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double q-learning," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 30, no. 1, 2016.

[17] R. Yurko, S. Ventura, and M. Horowitz, "nflwar: a reproducible method for offensive player evaluation in football," *Journal of Quantitative Analysis in Sports*, vol. 15, no. 3, pp. 163–183, 2019.

[18] Z. Thoutt, "Football player stats," https://www.kaggle.com/zynicide/nfl-football-player-stats.

Learning A Swarm Formation Policy using Deep Reinforcement Learning

Jasmina Rais
University of Alicante
San Vicente del Raspeig, Spain
jrm85@alu.ua.com

Fidel Aznar
University of Alicante
San Vicente del Raspeig, Spain
fidel@dccia.ua.es

***Abstract*—The purpose of this paper is to solve a swarm robotics task where a swarm of robots manages to follow the movements of a robot leader (that will develop an independent behaviour), and they maintain its formation while avoiding the environment obstacles. For this task, the problem will be modelled as a Markov Decision Process (MDP) and learned using Deep Reinforcement Learning with an evolutionary optimization CMA-ES strategy. Thus, we propose an fitness function based on the relative distance difference between the initial positions and the current positions formed by the robots. All the swarm individuals will have a positioning sensor relative to the leader robot and a laser sensor to be able to locate obstacles in the environment. Training and results analysis will be carried out in a realistic simulated environment. Once the system has been modelled and the learning process has been completed, we will study the robustness of the learned policy. The obtained results show that it is possible to learn a single swarm control policy able to follow a random behaviour of the leader, with different number of obstacles or noise in the local positioning system. In addition, we will compare the results obtained using genetic algorithms with those obtained by CMA-ES.**

***Index Terms*—Swarm robotics, Deep reinforcement learning, Evolutionary strategies, Swarm Intelligence**

I. Introduction

The purpose of this paper is find an optimal policy for a swarm robotics task, which will be executed homogeneously, since all robots will obtain the same behaviour, except for the leader. The desired behaviour is that a swarm of robots manages to follow the movements of the leader robot, which will have an independent behaviour, while maintaining a given shape with its initial positions and avoiding obstacles in the environment. In addition, it is important that the behaviour is robust to a random number of obstacles or to noise in the positioning sensor, as in a real environment there is no prior information about obstacles and sensors may have noisy readings.

Swarm robotics is a method of designing robotic behaviours in which robots work together in order to perform a task. This method is inspired by the self-organised behaviour of groups of individuals in the animal kingdom [1], [2]. Robots belonging to a swarm can perform a complex task by means of simple individual tasks. A behaviour can be modelled through design method based on existing behaviours in the animal kingdom or through automatic design method. The advantage of the latter method is that it reduces the effort required by the developer to model collective behaviour. In automatic design method, evolutionary algorithms are often used [3]–[5]. In addition, developmental strategies for reinforcement learning can be used to achieve a global maximum [6].

Autonomous navigation in multi-robot systems or collaborative robotics has been the subject of study in recent years [7], [8]. This task is very useful for exploring an environment without the limitations of using a single robot. In addition, it can be used to transport objects to the same location [9], as the swarm members follow the independent robot while maintaining the same shape. Therefore, the robots can transport objects to the place indicated by the leader robot. Another task that can be performed with this behaviour is the exploration of an environment [10], [11] because they can traverse the environment according to the shape of their initial positions without the need to partition the environment beforehand.

Throughout this article, other solved tasks similar to the swarm formation maintenance task will be shown. Furthermore, it will be explained how the solution of the main task has developed and how it has been solved after applying evolutionary strategies for reinforcement learning. In addition, simple genetic algorithms for reinforcement learning will also be used to evaluate the results obtained with CMA-ES. Algorithms based on genetic algorithms have been used in other papers to solve tasks similar to those in this paper. Therefore, the purpose of this comparison is to observe which evolutionary strategy offers an optimal policy for this task [12]. Finally, the results obtained in an environment with obstacles and random behaviour of the leader robot will be shown.

II. Related work

In this paper, evolutionary strategies for reinforcement learning have been used to model the behaviour of a swarm. For this reason, this section will describe some solved swarm robotics tasks and an autonomous obstacle navigation task.

First, this swarm must perform the task of following the movements of an independent robot while maintaining the shape that was formed by their initial positions. This task is similar in behaviour to the tasks of chain formation, coordinated movement and path planning. In addition, the robots must avoid obstacles in the environment by using a laser sensor to detect the distance of objects. Regarding the latter task, in the proposal of Huang et al. [13] uses Deep Q-learning to make

a robot wander through the environment without colliding with obstacles. The robot in this article detects obstacles using an infrared sensor and decides what action to take using a neural network trained with Deep Q-learning.

On the other hand, in the proposal by Sperati et al. [14] a genetic algorithm as an evolutionary strategy is used for exploration and navigation of a swarm in an unknown environment, that is solve the chain formation task. In addition, they use a model of the E-puck robot and the swarm's communication is via coloured LED lights. With this algorithm, the robots have been able to move to two specific areas to form a chain. Finally, in the path planning task, a swarm knows its initial position and the position that should be its final position, that is, the target position. To reach this position, they must decide which movements to make while moving together. In the proposal by Chien-Chou Lin et al. [12] genetic algorithms are used to achieve the ideal position by means of an optimal policy to maintain the shape of the swarm while following the leader robot. The reward function takes into account the current position the robots are in and the intermediate target position they should have at each instant.

Considering the state of the art and having seen the success of deep learning in several current applications, we wish to learn a dual policy capable of exploration and navigation of a swarm in an unknown environment using deep reinforcement learning. On the one hand it must be able to maintain the swarm formation and on the other hand it must allow agents to avoid environmental obstacles and return to their position in the formation. The fundamental advantage of end2end systems is to provide a parallel learning process for both policies, simplifying their design.

III. Problem design

Neural networks can be used to search for solutions to complex problems. One of the drawbacks of using learning techniques based on gradient information is that they generate little information from the environment in complex problems in robotics [15]. One of the advantages of using evolutionary strategies is that they are highly parallelisable, which speeds up the training process. They also have better exploration behaviour than some methods based on gradient information and they are robust to different hyperparameters [16]. However, regarding the exploitation and exploration in evolutionary strategies it has been concluded that the common view in EAs that the search space is explored by crossover/mutation operators, while exploitation is done by selection, is at least questionable [17].

The evolutionary strategy proposed in this paper is the covariance matrix adaptation strategy, CMA-ES. The evolutionary covariance matrix adaptation strategy, CMA-ES, is a strategy that takes the results of each generation and modifies the search space for the next generation. In each generation, μ y σ are adapted and the full covariance matrix is computed. The adaptation process is based on the N_{best} solutions and the input parameters ***x***, ***g*** the current generation and ***(g+1)*** is the next generation.

$$\mu_x^{(g+1)} = \frac{1}{N_{best}} \sum_{i=1}^{N_{best}} x_i \tag{1}$$

$$\sigma_x^{2,(g+1)} = \frac{1}{N_{best}} \sum_{i=1}^{N_{best}} (x_i - \mu_x^{(g)})^2 \tag{2}$$

This algorithm is one of the most popular for optimising reinforcement learning systems. The only drawback is the performance it can take if the number of parameters of the model to optimise is high, since the calculation of the covariance matrix is of order $O(N^2)$, although there are currently approximations that decrease it to $O(N)$.

In the main task of this paper, a swarm must always maintain the given initial shape while following the independent robot and dodging obstacles in the environment. This environment is unknown to the agents because they do not know where the obstacles are located or what path the lead robot will take. The possible actions that the leader robot will be able to execute are calculated through random values of linear and rotational velocity. At each execution step, each robot will observe its state $s_t \in S$ through a laser sensor with 180-degree angle coverage and 7 laser beams with a separation of 30-degree angle and a positioning sensor relative to the leader robot that indicates the X and Y position plus the angular orientation at which they are located. Both the models of the laser sensors and the differential actuators of the robots have been developed in a realistic manner, considering a physical MBot robot (https://www.makeblock.es/productos/mbotranger/). More specifically, all simulations have been performed using the Chipmunk physics simulation library (https://chipmunk-physics.net/), which has allowed us to greatly accelerate the training required to learn the policy.

Through a reward function, which will be explained later, each robot will have to choose which action to execute $a_t \in A$, which in this problem will be through the linear and rotational velocity that the motors will execute. The velocity values will be provided by the trained neural network. Thus, this task can be considered as a Markov Decision Process (MDP), where the observation space, S, is the state of the environment, that is the location of the obstacles or the position of the leader robot, and the action space, A, is the set of actions that the robots in the swarm can perform to avoid the obstacles and follow the leader robot's steps, in this case, the rotational and linear velocity.

On the other hand, a neural network has been used to map the information of the environment (see Fig. 1) with 10 inputs, 2 hidden layers and 2 output neurons, corresponding to the linear and rotational velocity. The activation function is tangent hyperbolic (tanh) and the weights of the neurons are determined by the strategies used for training.

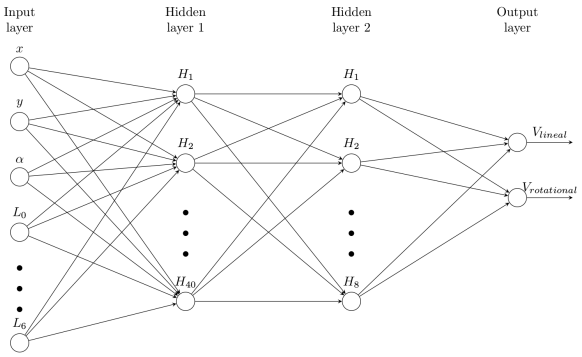


Fig. 1. Neural network used during training. It contains 10 input neurons. The inputs x and y are the difference between the initial and current distance of the swarm robot from the pivot robot in X and Y coordinates. The α input is the angular difference of the initial and final orientations of the swarm robot with respect to the pivot robot. The inputs L_0 to L_6 are the values obtained from the laser sensor beams. The input values are normalised. In addition, it contains 2 hidden layers of 40 and 8 neurons respectively and its activation function is tanh. Finally, there is the output layer which contains the linear and rotational velocity to be executed by the robot according to the information provided.

With regard to the reward function, it is proposed considering that the swarm forms an undirected graph where the robots are the nodes, and the edges are the distances between them. In this way, a table is obtained where the robots are in the rows and columns and the value of each element of the table is the corresponding distance. Then, the reward function is obtained by comparing the initial table of distances with that of the current instant, maintaining the margin of error of $\pm$ 5 cm. The distance is calculated using the Euclidean distance (Equation 3). Equations 4 and 5 are used to obtain the reward.

$$d = \sqrt{(X_{robot_i} - X_{robot_j})^2 + (Y_{robot_i} - Y_{robot_j})^2} \quad (3)$$

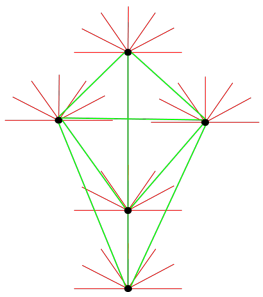


Fig. 2. Undirected graph formed by the swarm. The nodes are the robots and the edges, represented in green, are the distances between them.

$$\gamma = \sum_{i=1}^{N} \left(\sum_{j=1}^{N} |T_{o_{ij}} - T_{a_{ij}}| \right) \quad (4)$$

$$f(x) = \left(1 - \frac{\gamma}{\beta}\right) - \alpha \quad (5)$$

- $\boldsymbol{N}$: Number of robots in the swarm.
- $\boldsymbol{T_o}$: Table of distances between the robots at the initial step.
- $\boldsymbol{T_a}$: Table of distances between the robots at the current step.
- $\boldsymbol{\gamma}$: The result of the sum of the differences between the distance tables, that is the sum of all distances exceeding the 5 cm margin of error. It is used as a penalty for the solution obtained.
- $\boldsymbol{\beta}$: Maximum separation distance between the robots. It is used to normalise the result so that the penalty is between 0 and 1.
- $\boldsymbol{\alpha}$: Number of robot collisions with the obstacles.
- **f(x)**: Fitness function. Uses the total value obtained after penalties. In addition, a value that simulates the energy expenditure of each execution step is subtracted. The purpose of subtracting this energy is to increase the speed of the learning process.

The purpose of this reward function is to reward more the solutions that offer less difference in distances read with the positioning sensor with respect to those read at the initial instant of the execution. Therefore, the greater the difference in distance from the original position, the reward will be lower. The reward is obtained at each execution step.

Finally, this paper also uses simple genetic algorithms in order to find an optimal policy following the reinforcement learning scheme used with CMA-ES. Therefore, both strategies will use the same Markov modelling, the same reward function and the same neural network. In other proposals with tasks similar to the one proposed in this article, such as the chain training task or the path planning task, [12], [14], algorithms based on genetic algorithms are used, but with a mutation of genes in which the hyperparameters of the neural network are encoded. Therefore, in this proposal, both evolutionary strategies will be used for learning a behaviour by reinforcement learning. These strategies are CMA-ES, the one proposed in this paper, and simple genetic algorithms, whose evolutionary algorithm has been used in other papers.

IV. Experimentation

In this section we will explain how a swarm of agents was trained to move through an environment with obstacles following a robot acting as a leader robot. In addition, we will show the results of the behaviour obtained both in a simple scenario and in a more complex scenario with obstacles to avoid. Throughout the development of this section we will show a comparison of the strategy proposed in this article, CMA-ES, against simple genetic algorithms. The purpose of this comparison is to see what improvements CMA-ES has over genetic algorithms.

A. Experimental setup

As mentioned above, the robots receive information from the environment via the positioning sensor relative to the leader robot and the laser sensor. A simple environment with 5 robots and 2 obstacles with random positions near some of the robots in the swarm was used for training (see

Fig. 5). Obstacles have been simulated with 20 cm long square elements with a mass of 8 kg and no friction. Collisions are generated when the laser beam reading of the robots is less than 0.1 cm. The positions of the robots in the swarm at the beginning of the training are the same in each episode. The leader robot to be followed by the other robots in the swarm is positioned at the beginning of the swarm. The behaviour of this robot is pre-programmed with random movements. The movements it can execute are generated through random values of rotational and linear velocity, where the rotational velocity admits positive and negative values (to move left or right), while the linear velocity only admits positive values (to avoid backward movements). We design the environment with obstacles that do not hinder all robots, to achieve a flexible behaviour where the robots learn to follow the swarm leader while avoiding obstacles using the same learned policy.

The neural network training with CMA-ES and simple genetic algorithm, GA, has been obtained in 1000 generations, where each generation consisted of a population of 32 individuals. The elitism rate in the genetic algorithm used is 10%, this algorithm does not perform any mutation on individuals. Regard CMA-ES parameters, the initial value of $sigma$ used is $sigma = 0.1$.

B. Results

This section will show the results obtained after the learning process. Firstly, the results obtained with the CMA-ES and GA strategies will be compared in order to see which strategy offers better results. Regarding training results, in figure 3 we can see the convergence of both algorithms during training over 1000 generations. This figure shows that the CMA-ES strategy obtains the local maximum before the GA strategy.

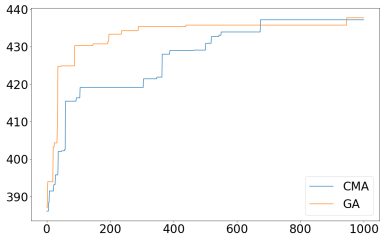
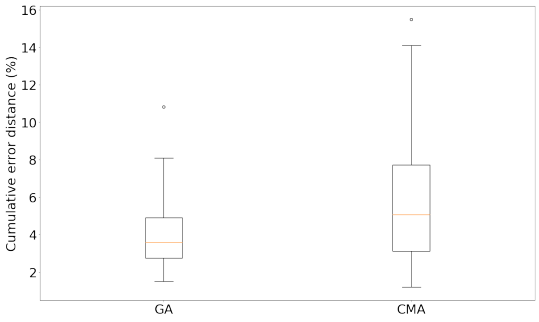
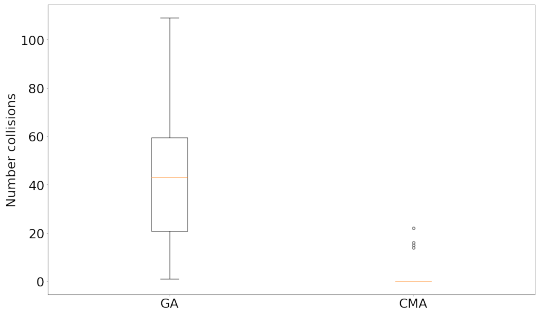


Fig. 3. Rewards obtained in process training over 1000 generations. The maximum possible global reward is 450 points, as they perform 450 execution steps. The training has been performed in an environment with 2 randomly located obstacles and a swarm consisting of 5 robots, where the leader is located in the upper central area and its route contains random movements.

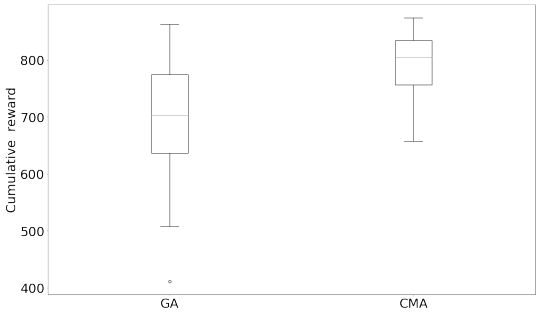
Secondly, the policies obtained by both strategies will be executed in the training environment consisting of 2 randomly located obstacles and a swarm of 5 robots. Although, in these tests, 900 moves were performed in each episode, therefore, the maximum reward is 900 points. After executing the policies obtained in 100 episodes, the results of the figure 4 are obtained. This figure shows a comparison of the two strategies with 3 metrics. The first metric is the average cumulative error distance relative to the total distance travelled of the swarm, the average is obtained over the 4 robots in the swarm executing the policy. The error distance is the distance between the actual position and the ideal position they should occupy with respect to the swarm leader robot. In this metric GA provides better results than CMA-ES, as the error distance is smaller. The second metric is the total number of collisions in the episode, where CMA-ES gives better results than GA, because the behaviour obtained collides fewer times with the obstacles. Finally, there is the third metric, the reward obtained during the episode. In this graph it can be seen that the CMA-ES policy also offers better results than GA, since, the elicited policy generates higher reward value. Therefore, we can conclude that the CMA-ES proposal provides better results than GA in 2 of the 3 metrics used.



(a) Error distance relative to the total distance travelled in 100 runs in the training environment.



(b) Total number of collisions during an episode in 100 runs.

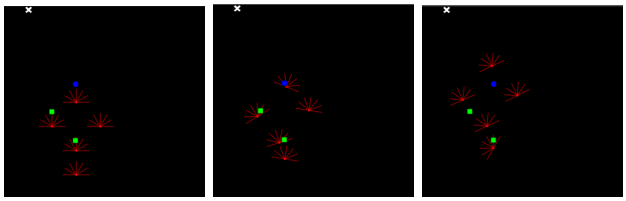


(c) Cumulative reward in 100 runs in the training environment.

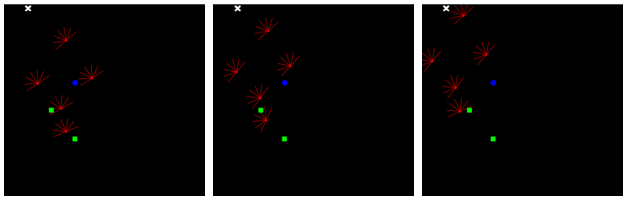
Fig. 4. Comparison of results obtained with the CMA-ES and GA strategies. The policies have been executed in the training environment in 900 execution steps. The global maximum reward of these executions is 900 points.

Now, focusing on the policy obtained by CMA-ES, we will evaluate if the policy is robust to noise or new obstacles. Figure 5 shows a sequence of the behaviour obtained by CMA-ES. In

this figure, it can be seen how the robots avoid the obstacles while following the leader robot, located in the upper central area.



(a) Step = 0 (b) Step = 200 (c) Step = 400



(d) Step = 600 (e) Step = 800 (f) Step = 1000

Fig. 5. Sequence of an execution of the learned behaviour where the robots in the swarm follow the leader robot while avoiding obstacles. Where (a) is the initial step and (f) is the final step. The initial and final location of the swarm are represented in each image of the sequence by a blue circle and a white cross respectively.

Figure 6 shows the results of the learned policy by CMA-ES on 100 runs with random leader moves. It can be seen that the policy develops an excellent result for the leader following task. In addition, a comparison has been made with the results obtained with different numbers of obstacles (see Fig. 7) in order to verify that the performance is robust against a variable number of obstacles. Behaviour learned in an obstacle-free environment generates less cumulative error, as the robots do not have to avoid any obstacles while following the leader robot. However, this error rate has a low value (less than 10 percent), so we consider that the behaviour has been learned successfully even increasing the number of obstacles with respect to those used in the training environment. Finally, in figure 8 we see several tests where Gaussian noise has been added to the relative location sensor of each robot to test its robustness to erroneous perceptions.

Based on the obtained results, we consider that the learned has developed the required behaviours for different scenarios with different numbers of obstacles and different paths of the leader robot. It is robust to noise in the location system, as slightly increasing the noise does not seriously impair the final behaviour. Thus, the robots do not collide with obstacles and follow the leader robot throughout the trajectory.



Fig. 6. Error distance relative to the total distance travelled in 100 runs in the training environment. The environment contains 2 obstacles of random locations and the paths of the lead robot are also random.

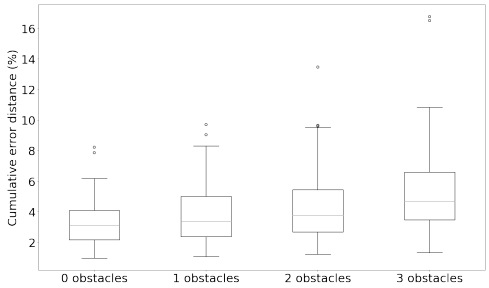


Fig. 7. Error distance relative to the total distance travelled in 100 runs (per parameter) with different number of obstacles. This graph shows the results obtained in environments with 0, 1, 2 and 3 obstacles with random location.

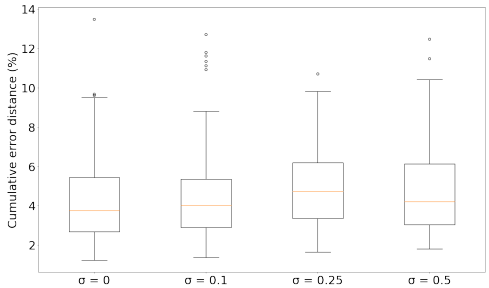


Fig. 8. Error distance relative to the total distance travelled in 100 runs (per parameter) with different noise values in the positioning system. This graph shows the results obtained in the training environment but applying Gaussian noise to the relative positioning system of the robots with respect to the leader robot. The values of the applied Gaussian noise have a normal distribution with $\mu = 0$ and $\sigma = 0$, $\sigma = 0.1$, $\sigma = 0.25$, $\sigma = 0.5$.

V. Conclusions

In this paper, a robotic swarm formation maintenance and obstacle avoidance policy has been obtained. A realistic modeling (both range sensors and differential actuators) has been used by means of a physical library that was compatible with the computational needs required by this type of training. The policy has been modeled as a Markov Decision Model (MDP) and learned using evolutionary strategies using the CMA-ES algorithm. Moreover, the results obtained with CMA-ES showed fewer collisions and higher reward at the end of the task than those obtained with simple genetic algorithms, GA,

for reinforcement learning, RL, whose evolutionary algorithm has been used in other papers with similar tasks.

The learned policy has been analyzed and very good results have been obtained, both in the training environments and in the robustness tests performed. More specifically, we have verified its robustness against noise present in the positioning system relative to the leader robot, against different number of obstacles and against random behaviour of the leader robot. Many simulations have been carried out with different scenarios where the location and number of obstacles is unknown to the swarm members and leader movement is random. We therefore consider that the learned homogeneous policy is able to develop the desired policy for all the swarm members.

As a future line, a physical environment with MBot robots will be designed to test the learned behaviour. Since we have tried to use the same actuators, sensors and physical conditions of the physical robots this step will allow us to verify what adjustments are required for the final policy deployment.

ACKNOWLEDGMENT

This project has been funded by the Ministry of Science, Innovation and Universities (Spain), project RTI2018-096219-B-I00. Project co-financed with FEDER funds.

REFERENCES

[1] Brambilla, M., Ferrante, E., Birattari, M., Dorigo M.: Swarm robotics: a review from the swarm engineering perspective. Swarm Intell. **7**, 1–41 (2013)

[2] Tan, Y., and Zheng, Z. Y. Research advance in swarm robotics. Defence Technology, 9, 18-39 (2013)

[3] SOYSAL, O., BAHEC, E., AHN, E.: Aggregation in Swarm Robotic Systems: Evolution and Probabilistic Control. TURKISH JOURNAL OF ELECTRICAL ENGINEERING & COMPUTER SCIENCES. **15**(2), 199–225 (2007)

[4] Trianni, V., Gro, R., Labella, T. H., ahin, E., Dorigo, M.: Evolving aggregation behaviors in a swarm of robots. Lecture notes in artificial intelligence. Advances in artificial life: 7th European conferenceECAL **2801**(4), 865–874 (2003)

[5] Jin, B. L. Generating collective foraging behavior for robotic swarm using deep reinforcement learning. Artificial Life and Robotics, 588-595. (2020)

[6] David Ha, A Visual Guide to Evolution Strategies, blog.otoro.net (2017).

[7] Cazangi, R. R., Von Zuben, F. J., Figueiredo, M. F. Autonomous navigation system applied to collective robotics with ant-inspired communication. In Proceedings of the 7th annual conference on Genetic and evolutionary computation, pp. 121-128. Association for Computing Machinery, New York, NY, USA(2005)

[8] T. Arai, E. Pagello, L. E. Parker. Guest editorial advances in multirobot systems. IEEE Transactions on Robotics and Automation, 18(5), 655-661, (2002). https://doi.org/10.1109/TRA.2002.806024.

[9] Campo, A., Nouyan, S., Birattari, M., Gro, R., Dorigo, M.: Enhancing cooperative transport using negotiation of goal direction. In Proceedings of the fifth international workshop on ant colony optimization and swarm intelligence (ANTS 2006), LNCS, Vol. 4150, pp. 365-366. Springer, Berlin (2006).

[10] Howard, A., Matari, M. J., Sukhatme, G. S.: Mobile sensor network deployment using potential fields: a distributed, scalable solution to the area coverage problem. Proceedings of the 2002 international symposium on distributed autonomous robotic systems (DARS 2002), 299–308 (2002)

[11] Ducatelle, F., Di Caro, C. P. G. A., Gambardella, L. M.: Self-organized cooperation between robotic swarms. Swarm Intelligence **5**(2), 73–96 (2011)

[12] Chien-Chou Lin, Kun-Cheng Chen, Po-Yuan Hsiao, Wei-Ju Chuang: Motion planning of swarm robots using potential-based genetic algorithm. International Journal of Innovative Computing, Information and Control **9**(1), 305–318 (2013)

[13] Bing-Qiang Huang, Guang-Yi Cao, Min Guo: Reinforcement Learning Neural Network to the Problem of Autonomous Mobile Robot Obstacle Avoidance. 2005 International Conference on Machine Learning and Cybernetics **1**, 85-89 (2005). https://doi.org/10.1109/ICMLC.2005.1526924

[14] Sperati, V., Trianni, V., Nolfi, S.: Self-organised path formation in a swarm of robots. Swarm Intelligence **5**, 97–119 (2011)

[15] Pagliuca, P., Milano, N., and Nolfi, S. Efficacy of Modern Neuro-Evolutionary Strategies for Continuous Control Optimization. arXiv preprint arXiv:1912.05239 (2019)

[16] Salimans, T., Ho, J., Chen, X., Sidor, S., and Sutskever, I. Evolution strategies as a scalable alternative to reinforcement learning. arXiv preprint arXiv:1703.03864 (2017)

[17] Črepinšek, M., Liu, S. H., Mernik, M. Exploration and exploitation in evolutionary algorithms: A survey. ACM computing surveys (CSUR), **45**(3), 1-33 (2013)

Generalized SMOTE: A universal generation oversampling technique for all data types in imbalanced learning

Carlos Cernuda
Electronics and Computing Dept.
Mondragon University
Arrasate-Mondragon, Spain
ccernuda@mondragon.edu

Daniel Reguera-Bakhache
Electronics and Computing Dept.
Mondragon University
Arrasate-Mondragon, Spain
dreguera@mondragon.edu

Aitor Aguirre
Electronics and Computing Dept.
Mondragon University
Arrasate-Mondragon, Spain
aaguirre@mondragon.edu

Mikel Iturbe
Electronics and Computing Dept.
Mondragon University
Arrasate-Mondragon, Spain
miturbe@mondragon.edu

Iñaki Garitano
Electronics and Computing Dept.
Mondragon University
Arrasate-Mondragon, Spain
igaritano@mondragon.edu

Urko Zurutuza
Electronics and Computing Dept.
Mondragon University
Arrasate-Mondragon, Spain
uzurutuza@mondragon.edu

***Abstract*— A common problem that arises when facing classification tasks is the class imbalance problem, which happens when one or more classes are heavily underrepresented compared to the rest, being usually those minority classes the ones of interest. A natural solution consists of correcting the imbalance by sampling methods, being Synthetic Minority Oversampling TEchnique (SMOTE) the most widely used method. In the same way as all other oversampling techniques, it relies on using distances/similarities in order to focus on the neighborhoods of minority samples in the synthetic samples generation procedure, thus it is meant for pure numerical data. Nevertheless, it is really common to collect categorical data or to discretize numeric attributes as a preprocessing step, being limited to random sampling approaches to correct imbalance. Some approaches have been proposed to deal with mixed-type data or pure categorical data, but they ignore part of the information of the samples or end up being almost random approaches.**

We propose GSMOTE, a generalization of SMOTE method, suitable for any data type. For the neighborhoods determination, the distance between samples is obtained by means of a transformation of Gower's General Similarity Coefficient into a novel General Distance Coefficient, in which the part corresponding to the way of measuring similarities between categories in categorical variables has been replaced by a recently presented similarity measure called Variable Entropy measure, inspired by Shannon's Entropy.

GSMOTE has been tested on six public imbalanced datasets, with different characteristics and imbalance levels.

***Index Terms*—Imbalanced Learning, Categorical Data, Mixed-Type Data, Oversampling Techniques, SMOTE**

I. Introduction

When facing a classification task is quite common to deal with datasets in which one or several classes are clearly underrepresented when compared to the rest. Such scenario is known as *imbalanced learning*. For instance, in real-world industrial or medical machine learning applications, such as fault detection or disease diagnosis, the classes of interest, respectively faulty and disease states, are pretty infrequent.

The underlying reasons for the imbalance are diverse. Industrial manufacturing processes [1] are highly optimized because no company could afford even moderate defective levels, thus usually less than 2% of the observations correspond to faulty states. In biomedical applications, the types of diseases that could benefit from machine learning models, for instance, are severe ones that are hard to detect by traditional methods with acceptable accuracy, which are not frequently happening [2] [3] [4] [5] [6]. In other cases, like in fraud detection in monetary transactions [7], the imbalance is due to the nature of the process itself, since most of the transactions are legal ones.

Independently of the root cause for the imbalance, the actions to prevent a classifier from predicting always the majority class (naïve classifier) are quite limited. Despite of the existence of certain taxonomies with more than two possibilities, we could include all into two:

(i) *Cost-sensitive learning* [8]. A cost-matrix is defined for the different errors that a classifier could make. Then, the classifier is trained in such a way that its goal is to minimize the global cost of all its errors. In this way, assigning a high cost to failing in classifying the minority class (or classes) would lead to models that are more accurate on predicting it (resp. them). The counterpart is that small improvements on the undesired errors would provoke an increase on the others that is much higher [9]. The trade-off between errors provided by our control over the cost-matrix is tricky to adjust, due to overfitting problems because of forcing the algorithms to focus on minority class samples during training [10].

(ii) *Sampling strategies*. The differences in the amount of samples from majority and minority classes is compensated

by increasing the amount minority class samples (oversampling) and/or decreasing the amount of majority class ones (undersampling). Because both have pros and cons [11] [12] [13], it is also common to opt for a hybrid approach (*mixed sampling*) [14] [15] [16] combining them.

In the case of *oversampling*, apart from randomly repeating some minority class samples, most of the approaches *generate* new synthetic minority class samples in the neighborhoods of the existing original ones. The most widely used algorithm is *synthetic minority oversampling technique (SMOTE)* [17]. In Smote, each synthetic sample, x_{new}, is randomly selected in the segment that connects one of the original minority class samples, x_i, with another minority class sample from its neighborhood, x_j. Unless we opt for random-based approaches, which usually perform poorly [15], we will focus on neighborhoods, thus distance-based approaches, meaning that we are limited to numerical features. Moreover, we could not discretize numerical features and then correct imbalance, but the other way around, leading to bins that are influenced by the synthetic data we have artificially introduced.

In the original SMOTE paper, the authors propose one adaptation for mixed type data, called *SMOTE Nominal Continuous* (SMOTE-NC), which assigns to every non-matching category of a categorical attribute a fixed distance equal to the median of the standard deviations of the numerical features for the minority class samples. Therefore, it uses only the information provided by the numeric part of the samples, leading to dangerous distances assumptions in case only a few of the attributes are numerical. Moreover, it ignores any information of the categorical attributes such as the amount of categories and their distribution.

Another adaptation is proposed for pure nominal features, called *SMOTE Nominal* (SMOTE-N), which uses a metric called Value Difference Metric [18] for characterizing distances/dissimilitudes. The way the synthetic samples are generated promotes keeping parts that are common with any of the neighbors'. This makes that synthetic samples coming from attributes with few values would be mainly copies, reducing to *random oversampling*.

In this paper, we propose an oversampling strategy that is suitable for all data types, i.e. pure categorical or numerical and mixed-type data, meant for binary problems. We have called it Generalized Synthetic Minority Oversampling TEchnique (GSMOTE) because we get original SMOTE when applied to pure numerical data, thus it could be considered as a generalization of SMOTE. The reason for considering the original SMOTE as an starting point is to see more clearly the potential of the approach. Nevertheless, the proposed generalization could be straightforwardly made from almost any variant.

In GSMOTE, the distances are calculated by means of a transformation of Gower's General Similarity Coefficient [19] into a distance coefficient. In this new General Distance Coefficient, the part related to categorical features has been replaced by a distance derived from the Variable Entropy (VE) measure [20], a similarity measure originally developed for hierarchical clustering of categorical data.

The creation of synthetic minority class samples from pairs of existing ones is decoupled for categorical and numerical parts of the samples. In both parts the same idea employed by SMOTE is maintained, i.e. randomly selecting one point in the segment that connects both original points by means of a convex combination, being adapted for the categorical parts keeping the concept of proximity to the extremes. Final synthetic samples are obtained by joining both parts.

The rest of the paper is organized as follow. In Section II, we introduce all previous works that are related to our method. In Section III, we present our contributions, consisting of a novel General Distance Coefficient (GDC), and the generalization of SMOTE algorithm, which uses the GDC. Section IV describes the datasets, experimental scheme and results achieved. Finally, in Section V we present conclusions and possible future works.

II. Premises

A. General Similarity Coefficient

In order to measure the similarity between two mixed-type data points, $\boldsymbol{x_i}$ and $\boldsymbol{x_j}$, Gower [19] defined the General Similarity Coefficient, given by

$$s_G\left(\boldsymbol{x_i}, \boldsymbol{x_j}\right) = \frac{1}{\sum_{k=1}^{d} w(x_{ik}, x_{jk})} \sum_{k=1}^{d} w(x_{ik}, x_{jk})\, s(x_{ik}, x_{jk}) \qquad (1)$$

where $s(x_{ik}, x_{jk})$ is a similarity component for the k-th attribute and $w(x_{ik}, x_{jk})$ is a binary constant with value 1 if the comparison between values for the k-th attribute is valid, and 0 otherwise.

The similarity components are defined differently depending on the types of attributes as follows

- For *quantitative* attributes, the similarity $s(x_{ik}, x_{jk})$ is defined as
$$s(x_{ik}, x_{jk}) = 1 - \frac{|x_{ik} - x_{jk}|}{R_k},$$
with R_k the range of the attribute.
- For *binary* attributes, the similarity is defined as
$$s_b(x_{ik}, x_{jk}) = \begin{cases} 1 & , \quad \text{if } x_{ik} = x_{jk} = true \\ 0 & , \quad \text{otherwise} \end{cases}$$
- For *nominal* or *categorical* attributes, the similarity $s(x_{ik}, x_{jk})$ is defined as
$$s(x_{ik}, x_{jk}) = \begin{cases} 1 & , \quad \text{if } x_{ik} = x_{jk} \\ 0 & , \quad \text{if } x_{ik} \neq x_{jk} \end{cases}$$

B. Variable Entropy similarity measure

Given a random variable X, taking values $x_1, \ldots, x_n$, *Shannon's Entropy* of X is defined as

$$H(X) = -\sum_{k=1}^{n} P(x_k) \log_b P(x_k)$$

where b is the base of the logarithm (usually taking values 2, e or 10). It could be interpreted as the uncertainty or the informativeness inherent to the possible outcomes of X, and it is

deeply related to the variability of the random variable. In fact, the maximum entropy happens when all values of the random variable are equally probable, i.e. maximum uncertainty.
With this interpretation in mind, we think of two different categorical variables, one with large variability and all values almost equally likely, and the other one with small variability, having one predominant category and the rest quite sparse. In such case, it is desirable that a match in any category of the first variable is interpreted as higher similarity than a match in the predominant category of the second, but lower similarity that a match in any of the sparse ones.
Accordingly, a similarity measure following the aforementioned philosophy should give higher weight to matches in the case of high variability, thus entropy.
Considering the sample relative frequencies of the categories as their probabilities, the similarity between categories is defined as [20]

$$S_k(x_{ik}, x_{jk}) = \begin{cases} -\dfrac{1}{\ln N_k} \displaystyle\sum_{u=1}^{N_k} p_u \ln p_u & , \quad \text{if } x_{ik} = x_{jk} \\ 0 & , \quad \text{if } x_{ik} \neq x_{jk} \end{cases} \tag{2}$$

Notice that in case of a match, it could theoretically take values in the unit interval $[0, 1]$, including 0. That null value could only be attained if one of the variables is constant. Nevertheless, in such case that category is not relevant for the similarity, thus null value seems reasonable. On the contrary, a value 1 is only possible if all categories in one of the variables are equally likely.
In order to determine the similarity between two samples $\boldsymbol{x_i}$ and $\boldsymbol{x_j}$, we define the *Variable Entropy* (VE) similarity measure as

$$S_{ve}\left(\boldsymbol{x_i}, \boldsymbol{x_j}\right) = \frac{\displaystyle\sum_{k=1}^{d} S_k(x_{ik}, x_{jk})}{d} \tag{3}$$

Again, the similarity measure takes values in $[0, 1]$, being the extreme values accessible if and only if all d averaged values take that exact extreme value.

C. Synthetic Minority Oversampling TEchnique

Synthetic Minority Oversampling TEchnique [17], briefly described above, is an oversampling technique that, in its original version, generates synthetic minority class samples inside randomly selected minority class samples' neighborhoods, which are obtained by *k*-Nearest Neighbors algorithm with a predefined *k*. The new synthetic minority class samples are determined by randomly sampling points in the segment that connects the initial sample with one of its *k* neighbors.
If we denote by $\boldsymbol{x_i}$ the central original minority class sample, and by $\boldsymbol{x_{n_1}}, \ldots, \boldsymbol{x_{n_k}}$ its K minority class nearest neighbors, then one of the K neighbors is randomly selected (let us denote it by $\boldsymbol{x_j}$). Mathematically, the segment joining two points is given by their convex linear combination. In this case, it corresponds to

$$\boldsymbol{x_{new}} = \lambda \boldsymbol{x_j} + (1 - \lambda)\boldsymbol{x_i}, \quad \text{with } \lambda \in [0, 1] \tag{4}$$

The selection of one single point is obtained by sampling one λ value. Notice that the two boundary λ values would replicate the extreme points of the segment.

III. Our approach

A. General Distance Coefficient

On the basis of Gower's General Similarity Coefficient, and taking into account the relationship between dissimilarity and distance, we can derive the following *General Distance Coefficient*, given by

$$d_{gdc}\left(\boldsymbol{x_i}, \boldsymbol{x_j}\right) = \left(\frac{1}{\displaystyle\sum_{k=1}^{d} w(x_{ik}, x_{jk})} \sum_{k=1}^{d} w(x_{ik}, x_{jk})\, d^2(x_{ik}, x_{jk}) \right)^{1/2} \tag{5}$$

where $d^2(x_{ik}, x_{jk})$ is a squared distance component for the k-th attribute and $w(x_{ik}, x_{jk})$ is a binary constant with value 1 if the comparison between values for the k-th attribute is valid, and 0 otherwise.
The distance components are defined differently depending on the types of attributes as follows

- For *continuous* and *ordinal* attributes, the distance $d(x_{ik}, x_{jk})$ is defined as

  $$d(x_{ik}, x_{jk}) = \frac{|x_{ik} - x_{jk}|}{R_k},$$

  with R_k the range of the attribute.
- For *quantitative* attributes, $d(x_{ik}, x_{jk})$ is defined as

  $$d(x_{ik}, x_{jk}) = |x_{ik} - x_{jk}|$$

  If we standardize $x_{\bullet k}$ as

  $$x^*_{\bullet k} = \frac{x_{\bullet k} - \mu_k}{\sigma_k},$$

  where μ_k and $sigma_k$ are, respectively, the mean and standard deviation of the k-th attribute, it could also be normalized, such that

  $$d(x_{ik}, x_{jk}) = \frac{\left|x^2_{ik} - x^2_{jk}\right|}{\sigma_k}.$$

- For *binary* attributes, the distance is defined, also following VE philosophy, as

  $$d(x_{ik}, x_{jk}) = \begin{cases} 1 - \dfrac{1}{\ln 2} \displaystyle\sum_{u=1}^{2} p_u \ln p_u & , \quad \text{if } x_{ik} = x_{jk} = true \\ 1 & , \quad \text{otherwise} \end{cases}$$

  Notice that null distance is reachable if and only if

  $$p_1 = P(true) = 1/2 = P(false) = p_2$$

- For *nominal* or *categorical* (not *binary*) attributes, the distance $d(x_{ik}, x_{jk})$ is defined as the dissimilarity obtained from the VE similarity measure

  $$d(x_{ik}, x_{jk}) = \begin{cases} 1 - \dfrac{1}{\ln N_k} \displaystyle\sum_{u=1}^{N_k} p_u \ln p_u & , \quad \text{if } x_{ik} = x_{jk} \\ 1 & , \quad \text{if } x_{ik} \neq x_{jk} \end{cases}$$

  where $N_k \geq 3$.

The reason for the separation into *continuous and ordinal* and the *rest of quantitative* attributes in the way of normalizing, avoiding the range for the latter, is the lack of determination of the variability using only the information provided by the maximum and minimum values in such case. Instead, the standard deviation is more reasonable [21].

B. Generalized Synthetic Minority Oversampling TEchnique

The Generalized Synthetic Minority Oversampling TEchnique (GSMOTE) algorithm comprises the following steps:

1) Input:

(i) A dataset D, with dimensionality d (i.e. number of input features) and a binary target class vector, given by

$$\{(\boldsymbol{x_i} = (x_{i1}, \ldots, x_{id}), y_i), i = 1, \ldots, m\}$$

where m_p samples correspond to the positive (minority) class and m_n to the negative (majority) class, with $m = m_p + m_n$; and where $d_c \geq 0$ features are categorical/binary and $d_n \geq 0$ are numerical, so that $d = d_c + d_n$.

(ii) A predefined *number of neighbors* to be considered for synthetic samples generation (default $K = 5$)

(iii) A predefined *maximum permitted imbalance ratio* (default $\beta = 0.1$).

2) Procedure:

(i) Calculation of the *degree of imbalance*, which belongs to $(0, 1]$,

$$b = {}^{m_p}/_{m_n} \tag{6}$$

(ii) Calculation of the *amount of synthetic minority samples*, m_s, needed to fulfill the β limitation, that is

$$m_s = \lceil m_n(1 - \beta) - m_p \rceil \tag{7}$$

where $\lceil z \rceil$ denotes the ceiling function, i.e. the smallest integer bigger or equal than z. Notice that, if no imbalance is permitted, then $\beta = 0$ and, therefore, $m_s = m_n - m_p$, as expected.

(iii) For each sample $\boldsymbol{x_i}$ we calculate its K nearest neighbors, using the *general distance coefficient* in Equation 5.

(iv) Assuming, without loss of generality, that the first d_c samples are categorical/binary and the last d_n are numerical, and denoting by P the subset of minority class samples

$$P = \{\boldsymbol{x_i} = (x_{i1}, \ldots, x_{id}), i = 1, \ldots, m_p\} \tag{8}$$

For each sample

$$\boldsymbol{x_i} = (\underbrace{x_{i1}, \ldots, x_{id_c}}_{\text{Cat/Bin}}, \underbrace{x_{i(d_c+1)}, \ldots, x_{id}}_{\text{Numerical}}) \tag{9}$$

For operative reasons in the following path calculation step, we have kept together categorical and binary attributes in the separation. Nevertheless, they have been treated differently in the distance calculation.

(v) We repeat G times the following process

1) Randomly select one sample $\boldsymbol{x_i}$ from P.
2) Consider its neighborhood and randomly select one of its neighbors, $\boldsymbol{x_j}$. Split both samples in their categorical/binary and numerical parts. We denote them by $\boldsymbol{x_i^c}$ and $\boldsymbol{x_j^c}$, for categorical/binary parts, and by $\boldsymbol{x_i^n}$ and $\boldsymbol{x_j^n}$ for numérical.
3) Calculate the path from the categorical/binary part of $\boldsymbol{x_i}$

$$\boldsymbol{x_i^c} = (x_{i1}, \ldots, x_{id_c})$$

to the one of $\boldsymbol{x_j}$

$$\boldsymbol{x_j^c} = (x_{j1}, \ldots, x_{jd_c})$$

ordering the samples from closest to farthest from $\boldsymbol{x_i}$, according to the *Manhattan distance*. In this case, with our notation, that distance is defined as

$$d_M(\boldsymbol{x_i^c}, \boldsymbol{x_j^c}) = \sum_{k=1}^{d_c} |x_{ik} - x_{jk}|$$

As an example, consider that $\boldsymbol{x_i^c} = (1, 1, 3, 2)$ and $\boldsymbol{x_j^c} = (1, 3, 3, 1)$. The steps in the path, including all Manhattan distances to $\boldsymbol{x_i^c}$, would be

$$\begin{array}{rcrcll} \boldsymbol{x_i^c} & = & \boldsymbol{x_{s_0}^c} & = & (1,1,3,2) & \Rightarrow \; d_M = 0 \\ & & \boldsymbol{x_{s_1}^c} & = & (1,2,3,2) & \Rightarrow \; d_M = 1 \\ & & \boldsymbol{x_{s_2}^c} & = & (1,1,3,1) & \Rightarrow \; d_M = 1 \\ & & \boldsymbol{x_{s_3}^c} & = & (1,2,3,1) & \Rightarrow \; d_M = 2 \\ & & \boldsymbol{x_{s_4}^c} & = & (1,3,3,2) & \Rightarrow \; d_M = 2 \\ \boldsymbol{x_j^c} & = & \boldsymbol{x_{s_5}^c} & = & (1,3,3,1) & \Rightarrow \; d_M = 3 \end{array}$$

4) Sample a random λ in $[0, 1]$, and then
4.1. Apply Eq. 4 to the numeric parts, obtaining the new synthetic numerical part $\boldsymbol{x_{new}^n}$, as in original SMOTE algorithm.
4.2. If we denote by n_s the amount of intermediate steps in the path, and we define $\rho = \frac{1}{n_s+1}$, we can define the following partition of the unit interval

$$[0,1] = \underbrace{\left[0, \frac{1}{2}\rho\right]}_{I_0} \cup \bigcup_{v=1}^{n_s} \underbrace{\left(\frac{2v-1}{2}\rho, \frac{2v+1}{2}\rho\right]}_{I_v} \cup \underbrace{\left(\frac{2n_s+1}{2}\rho, 1\right]}_{I_{n_s+1}} \tag{10}$$

Then, we obtain the new synthetic categorical/binary part by selecting $\boldsymbol{x_{new}^c} = \boldsymbol{x_{s_w}^c}$, where w is so that $\lambda \in I_w$. In the previous example, $n_s = 4$, $\rho = {}^1/_5$, thus the partition of the unit interval would be

$$\underbrace{\left[0, \frac{1}{10}\right]}_{I_0} \cup \underbrace{\left(\frac{1}{10}, \frac{3}{10}\right]}_{I_1} \cup \underbrace{\left(\frac{3}{10}, \frac{1}{2}\right]}_{I_2} \cup \underbrace{\left(\frac{1}{2}, \frac{7}{10}\right]}_{I_3} \cup \underbrace{\left(\frac{7}{10}, \frac{9}{10}\right]}_{I_4} \cup \underbrace{\left(\frac{9}{10}, 1\right]}_{I_5}$$

For instance, if $\lambda = 0.47 \in \left(\frac{3}{10}, \frac{1}{2}\right] = I_2$, then $\boldsymbol{x_{new}^c} = \boldsymbol{x_{s_2}^c} = (1, 1, 3, 1)$.

5) Get the new synthetic sample $\boldsymbol{x_{new}}$ by joining both new parts $\boldsymbol{x_{new}^c}$ and $\boldsymbol{x_{new}^n}$. Under our assumptions

$$\boldsymbol{x_{new}} = (\boldsymbol{x_{new}^c}, \boldsymbol{x_{new}^n}) \tag{11}$$

(vi) We add the new G synthetic minority samples to the dataset.

TABLE I
SUMMARY INFORMATION OF THE UCI DATASETS USED.

Name	Target	Imb. ratio	# Samples	# Feat.
Ecoli	imU	8.6 : 1	336	7 num
Pen Dig.	Number 5	9.4 : 1	10992	16 num
Wine Qual.	Quality ≤ 4	26 : 1	4898	11 num
Sick Euth.	sick_euthyroid	9.8 : 1	3163	6 num, 36 cat
Car Eval.	good, vgood	12 : 1	1728	6 cat
Solar Fl.	Moderate = 0	19 : 1	1389	10 cat

TABLE II
RESULTS, MEAN $\pm$ STD WITH TEN REPETITIONS, FOR DIFFERENT PERCENTAGES OF CATEGORICAL/BINARY DATA IN ECOLI, PEN DIGITS AND WINE QUALITY DATASETS.

Perc.	Method	Ecoli	Pen Dig.	Wine Qual.
25%	No	.9123 $\pm$ .0140 †	.99984 $\pm$ 10^{-5}	.8427 $\pm$ .0027 †
	SM-NC	.9926 $\pm$ .0019	.99981 $\pm$ 10^{-5}	.9843 $\pm$ .0008 †
	GSMOTE	**.9970 $\pm$ 0.0033**	**.99993** $\pm$ 10^{-6}	**.9942 $\pm$ .0008**
50%	No	.9176 $\pm$ .0107 †	.99984 $\pm$ 10^{-5}	.8559 $\pm$ .0029 †
	SM-NC	.9872 $\pm$ .0116	.99988 $\pm$ 10^{-5}	.9865 $\pm$ .0017
	GSMOTE	**.9931 $\pm$ .0014**	**.99991** $\pm$ 10^{-5}	**.9946 $\pm$ .0007**
75%	No	.9354 $\pm$ .0098 †	.99984 $\pm$ 10^{-5}	.8514 $\pm$ .0032 †
	SM-NC	.9822 $\pm$ .0087 †	**.99990** $\pm$ 10^{-5}	.9933 $\pm$ .0007
	GSMOTE	**.9934 $\pm$ .0021**	**.99990** $\pm$ 10^{-5}	**.9960 $\pm$ .0004**

IV. EXPERIMENTS

A. Experimental scheme

We have considered six datasets from UCI repository [22], in the way they have been proposed in [23]. Table I summarizes the information about them.

For validation purposes we will compare the performance of GSMOTE with the original imbalanced sets, i.e. no imbalance correction, in the case of categorical/binary data, and also with SMOTE-NC in mixed-type data.

In the latter, in order to check the behavior of the algorithms for different ratios of categorical vs numerical attributes, we have considered the three pure numerical datasets (Ecoli, Pen Digits, and Wine Quality [24]). For each of them, we have randomly selected 25%, 50% and 75% of their attributes, and we have discretized them by means of the Chi2 discretization algorithm [25]. In order to mitigate possible deviations because of the attribute selection by chance, we have performed each ten times. In the former, we have considered all six datasets described in Table I, in which all numerical attributes have also been discretized by Chi2 algorithm.

The algorithm employed for the tests is random forest (RF) [26], because it is a powerful algorithm that is pretty efficient for mixed-type data, due to be based on decision trees. For the selection of the hyperparameters of the model, we have performed an exhaustive grid search, varying the number of trees (in $\{50, 100, 200\}$), the maximum depth of each tree (in $\{5, 10, 20, \infty\}$) and the amount of attributes visible in each splitting node (in $\{d, \sqrt{d}\}$), coupled with a 10-folds cross validation strategy, using the area under the receiver operator curve (AUC) for scoring because it is robust against the effect of imbalance in absence of a cost-sensitive approach [27]. Moreover, as random forest is a stochastic method, so we have repeated each procedure 10 times. Therefore, for each imbalance learning method, we have made 720 trials per dataset, including the original ones.

B. Results and discussion

The results discussed are presented as follows. First, in the case of mixed-type data, we report the results of the average score of the 10 repetitions of RF, for the best hyperparameters of the grid in each case in Table II. Last, in the case of pure categorical/binary data, the results are presented in Table III. In both cases, statistical significance tests have been carried out by means of Wilcoxon signed-rank test [28], being indicated in the tables in case the null hypothesis has been rejected (meaning a significant difference) when compared with the best (in bold).

In the case of Ecoli, we can appreciate that the performance of both approaches is around 10% better in average than the imbalanced dataset. Despite it cannot be said that GSMOTE behaves much better, it is slightly better most of the times, independently of the percentage of categorical attributes. One should notice that the dimensionality of this dataset is low, so there is not much margin to appreciate the influence of the type of estimation made by SMOTE-NC based on the standard deviations, except perhaps for 75%. In fact, this is the case in which the difference of the average AUCs is higher (2.5 times higher than for 25%). For pure categorical data, we can appreciate that the AUC score is also high for 100% categorical attributes, but lower than the averages of the rest of lower percentages.

When it comes to mixed data in Pen Digits dataset, the margin for improvement is really low because the AUC is very high even without correcting the imbalance. Nevertheless, this could be seen as a challenge both approaches have passed. Again, GSMOTE overtakes SMOTE-NC in single wins, but this time the highest difference happens with the lowest proportion of categorical features. Without an exhaustive study of the nature and characteristics of the datasets, out of the scope of this paper, it is hard to find a reason for it. The tendency remains the same for full categorical data, including the tight margin for improvement (see Table III).

Now, on mixed-type data for Wine Quality dataset, we could say that GSMOTE is clearly the best algorithm, winning every single time, and rising up the AUC score around 15% with respect to the original imbalanced data. It is also noticeable that, in all three cases, the higher the percentage of categorical features the better the performance. This is not maintained in the case of pure categorical data, as can be seen in Table III.

Finally, Table III presents the results on pure categorical/binary data for all datasets. In all cases, it is beneficial to perform the imbalance correction. Apart from the three numerical datasets, commented before, Eurothyroids is a clear example of the possibility of getting a mixed-type set and discretize all numerical attributes (6 up to 36 in this case) before correcting the imbalance, leading to successful results. Finally, CarEval

TABLE III
RESULTS FOR PURE CATEGORICAL/BINARY DATA.

Method	Ecoli	Pen	Wine	Sick	Car	Solar
No	.9477 †	.9998	.8602 †	.9835 †	.9871	.7919 †
GSMOTE	**.9926**	**.9999**	**.9517**	**.9950**	**.9966**	**.9030**

and SolarFlare, pure categorical originally, benefit as well from the correction (14% in the case of SolarFlare).

V. CONCLUSIONS

We have presented Generalized SMOTE, a prototype-generation oversampling method that could be employed in pure numerical and categorical, as well as in mixed-type data. In the particular case of pure numerical data, it reduces to the original SMOTE.

The main advantage over other existing techniques is the better use of the inherent information, because it takes into account the amount of categories and their distribution in the minority class subspace, in order to assign the distances between samples in the neighborhood search. Moreover, it decouples the prototype generation in categorical and numerical parts, being able to keep the notion of proximity implicit in the way SMOTE generates synthetic samples in the segments connecting original ones.

The results obtained in six public diverse imbalanced datasets support GSMOTE as a powerfull method.

ACKNOWLEDGMENT

This work has been partially funded by the project Semantic Knowledge Integration for Content-Based Spam Filtering (TIN2017-84658-C2-2-R) from the Spanish Ministry of Economy, Industry and Competitiveness (SMEIC), State Research Agency (SRA) and European Regional Development Fund (ERDF), and by the project REMEDY - REal tiME control and embeddeD securitY from the Department of Economic Development and Infrastructures of the Basque Government under the grant agreement KK-2021/00091. It has been developed by the intelligent systems for industrial systems group supported by the Department of Education, Language policy and Culture of the Basque Government.

REFERENCES

[1] Cernuda C.: On the Relevance of Preprocessing in Predictive Maintenance for Dynamic Systems. In: Lughofer E., Sayed-Mouchaweh M. (eds) Predictive Maintenance in Dynamic Systems. Springer, **Chapter 3**, 53–93 (2019)

[2] Davis D.N., Nguyen T.T.T.: Generating and Verifying Risk Prediction Models Using Data Mining (A Case Study from Cardiovascular Medicine). 57th Annual Congress of the European Society for Cardiovascular Surgery (ESCVS), Barcelona, Spain (2008)

[3] Mar J., Gorostiza A., Arrospide A. et al.: Estimation of the epidemiology of dementia and associated neuropsychiatric symptoms by applying machine learning to real-world data. Journal of Psychiatry and Mental Health (2021). DOI: *https://doi.org/10.1016/j.rpsm.2021.03.001*

[4] Laza R., Pavon R., Reboiro-Jato M., Fdez-Riverola F.: Evaluating the effect of unbalanced data in biomedical document classification. Journal of Integrative Bioinformatics **8**(3), 1–13 (2011)

[5] Rahman M.M., Davis D.N.: Addressing the Class Imbalance Problem in Medical Datasets. International Journal of Machine Learning and Computing **3**(2), 224–228 (2013)

[6] Mar J., Gorostiza A., Ibarrondo O. et al.: Validation of random forest machine learning models to predict dementia-related neuropsychiatric symptoms in real-world data. Journal of Alzheimer's Disease **77**(2), 855–864 (2020)

[7] Phua, C., Alahakoon, D.: Minority report in fraud detection: Classification of skewed data. ACM SIGKDD Explorations Newsletter **6**(1), 50–59 (2004)

[8] Thai-Nghe N., Gantner Z., Schmidt-Thieme L.: Cost-sensitive learning methods for imbalanced data. The 2010 International Joint Conference on Neural Networks (IJCNN), Barcelona Spain, 1–8 (2010)

[9] Attenberg J., Ertekin S.: Class Imbalance and Active Learning. In: He H., Ma Y. (eds) Imbalanced Learning: Foundations, Algorithms, and Applications. John Wiley & Sons, Chapter 6, 101–149 (2013)

[10] Fernández A., García S., Galar M., Prati R.C., Krawczyk B., Herrera F.: Cost-Sensitive Learning. In: Learning from Imbalanced Data Sets. Springer, Cham, Chapter 4, 63–78 (2018)

[11] Chawla, N.V.: C4.5 and Imbalanced Data sets: Investigating the Effect of Sampling Method, Probabilistic Estimate, and Decision Tree Structure. Proceedings of the ICML'03 Workshop on Learning from Imbalanced Data sets, Washington, DC, USA (2003)

[12] Drummond, C., Holte, R.: C4.5, class imbalance, and cost sensitivity: Why undersampling beats over-sampling. Proceedings of the ICML'03 Workshop on Learning from Imbalanced Data Sets, Washington, DC, USA (2003)

[13] Maloof, M.: Learning when data sets are imbalanced and when costs are unequal and unknown. Proceedings of the ICML'03 Workshop on Learning from Imbalanced Data Sets, Washington, DC, USA (2003)

[14] Kubat, M., Matwin, S.: Addressing the Curse of Imbalanced Training Sets: One Sided Selection. Proceedings of the Fourteenth International Conference on Machine Learning, 179–186, Nashville, Tennesse. Morgan Kaufmann (1997)

[15] Japkowicz, N.:The Class Imbalance Problem: Significance and Strategies. Proceedings of the International Conference on Artificial Intelligence (IC-AI'2000): Special Track on Inductive Learning, 111–117, Las Vegas, Nevada, USA (2000)

[16] Batista, G.E.A.P.A., Prati, R.C., Monard, M.C.: A study of the behavior of several methods for balancing machine learning training data. ACM SIGKDD Explorations Newsletter **6**(1), 20–29 (2004)

[17] Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: SMOTE: Synthetic Minority Oversampling TEchnique. Journal of Artificial Intelligence Research **16**, 321–357 (2002)

[18] Cost S., Salzberg S.: A Weighted Nearest Neighbor Algorithm for Learning with Symbolic Features. Machine Learning **10**(1), 57–78 (1993)

[19] Gower, J.C.: A General Coefficient of Similarity and Some of Its Properties. Biometrics **27**(4), 857–871 (1971)

[20] Sulc Z., Rezankova H.: Comparison of Similarity Measures for Categorical Data in Hierarchical Clustering. Journal of Classification **35**(1), 58–72 (2019)

[21] Wishart D.: k-Means Clustering with Outlier Detection, Mixed Variables and Missing Values. In: Schwaiger M. and Opitz O. (edts) Exploratory Data Analysis in Empirical Research, Springer, 216–226 (2003)

[22] Dua D., Graff C.: UCI Machine Learning Repository. Irvine, CA: University of California, School of Information and Computer Science (2019). *https://archive.ics.uci.edu/ml/*

[23] Zejin D.: Diversified Ensemble Classifiers for Highly Imbalanced Data Learning and their Application in Bioinformatics. Dissertation, Georgia State University (2011)

[24] Cortez P., Cerdeira A., Almeida F., Matos T., Reis J.: Modeling wine preferences by data mining from physicochemical properties. Decision Support Systems **47**(4), 547–553 (2009)

[25] Liu H., Setiono R.: Feature selection via discretization. IEEE Transactions on Knowledge and Data Engineering **9**(4), 642–645 (1997)

[26] Breiman, L.: Random forests. Machine Learning, vol. 45(1), pp. 5 – 32 (2001)

[27] Ferri C., Flach P., Orallo J., Lachice N. (edts). ECAI'2004 First Workshop on ROC Analysis in Artificial Intelligence (2004)

[28] Wilcoxon F.: Individual comparisons by ranking methods. Biometrics Bulletin **1**(6), 80–83 (1945)

Deep Reinforcement and Imitation Learning for Self-Driving Tasks*

*Note: The full contents of this paper have been published in the volume Lecture Notes in Artificial Intelligence 12882 (LNAI 12882)

Sergio Hernández-García, Alfredo Cuesta-Infante
Universidad Rey Juan Carlos
Madrid, Spain
{sergio.hernandez, alfredo.cuesta}@urjc.es

***Abstract*—In this paper we train four different deep reinforcement and imitation learning agents on two self-driving tasks. The environment is a driving simulator in which the car is *virtually* equipped with a monocular RGB-D camera in the windshield, has a sensor in the speedometer and actuators in the brakes, accelerator and steering wheel. In the imitation learning framework, the human expert sees a photorealistic road and the speedometer, and acts with pedals and steering wheel. To be efficient, the state is a representation in the feature space extracted from the RGB images with a variational autoencoder, which is trained before running any simulation with a loss that attempts to reconstruct three images, the same RGB input, the depth image and the segmented image.**

***Index Terms*—Self-driving, Imitation Learning, Reinforcement Learning.**

DOI: https://doi.org/10.1007/978-3-030-85713-4_7

A Proposal to Integrate Deep Q-Learning with Automated Planning to Improve the Performance of a Planning-based Agent*

*Note: The full contents of this paper have been published in the volume Lecture Notes in Artificial Intelligence 12882 (LNAI 12882)

Carlos Núñez-Molina, Ignacio Vellido, Vladislav Nikolov-Vasilev, Raúl Pérez, Juan Fdez-Olivares
Dept. of Computer Science and AI
University of Granada
Granada, Spain
{ccaarlos,ignaciove}@correo.ugr.es,vladis890@gmail,{fgr,faro}@decsai.ugr.es

***Abstract*—In this work we propose an architecture which learns to select subgoals with Deep Q-Learning in order to decrease the load of a planner when faced with scenarios with tight time restrictions, such as online execution systems. We have trained this architecture on a video game environment used as a standard testbed for intelligent systems applications. We experiment with different values of the discount rate γ and show the importance of long-term thinking when selecting subgoals. We also compare our approach against a classical planner and show how it is able to greatly reduce time requirements, although obtaining plans with 25% more actions on average. We conclude our approach is competitive with a classical planner and presents better generalization properties than most Reinforcement Learning algorithms when applied to new levels of the same game.**

***Index Terms*—Automated planning, Goal reasoning, Deep Q-learning.**

DOI: https://doi.org/10.1007/978-3-030-85713-4_3

Ontologies for run-time self-adaptation of mobile robotic systems

Esther Aguado, Ricardo Sanz and Claudio Rossi
UPM Autonomous Systems Laboratory, Centre for Automation and Robotics UPM-CSIC,
Universidad Politécnica de Madrid, Spain

***Abstract*—The fundamental task of a systems engineer is to map user needs to product realizations that robustly fulfill their missions in real, complex environments plagued with uncertainty. Real operational conditions of systems may deviate from the assumptions made in the system design process, compromising the system's ability to fulfill the mission. Simple controllers can cope with some deviations, but they fail when the deviations are large or unexpected. One approach to overcome this challenge is by providing the system with more flexible adaptation strategies that incorporate the systems engineer knowledge in the form of run-time system self-knowledge that empowers self-management. This paper specifically focuses on the use of ontologies as a vehicle to leverage system design and implementation knowledge during run-time operation for system self-adaptation. The use of system-centered ontologies within a system architectural model allows the explicit definition, allocation, and linkage of system requirements, design decisions, system realizations, and run-time information. This knowledge can then be used by a reasoner during system operation to evaluate different designs at several performance dimensions degrees considering the actual situation of the running system. This paper addresses these issues in the domain of mobile robotics. We analyze previous experiences and define future lines for the effective use of ontologies to augment the resilience of autonomous mobile robot systems.**

I. Introduction

An artificial system is an aggregate of interconnected elements engineered to fulfill a need [1]. This need is commonly satisfied by driving some changes of value to the user/system environment. In some cases, the value can only be delivered by systems of high complexity and their construction requires a knowledge-intensive, well-orchestrated engineering process.

Systems engineering (SE) is defined by INCOSE as a multidisciplinary approach to the design, realization, and deployment of a system. SE activities cover the whole system life-cycle, including exploitation, technical management, and system decommission [2]. Using SE, we can build systems of enormous complexity as is the case of autonomous robots.

However, once a system is deployed to be used, the environment in which it operates may suffer variations from its original configuration. Moreover, the system itself may suffer deviations from its initial design due to drift or subsystem failures. Both the assumptions concerning the environment and the stability of system characteristics may be violated in the changing dynamics of the real world. These violations are usually addressed by means of supervisory control, but this implies the continuous involvement of humans in the system —with associated costs, risks, and limitations.

In the case of building autonomous systems, there are no humans on top, and the engineer shall endow the system itself with the capability of addressing these expected and unexpected contingencies using its own resources and capabilities. Among these capabilities, the work described in this paper addresses the capability for *self-adaptation*.

Self-adaptation is a powerful, open strategy to augment the autonomy levels of complex systems in complex environments; and is of maximal importance for environments with a high level of uncertainty. It is enormously powerful because it may target changes in the system architectural structure, in the executed behaviour, or both.

Structural self-adaptation consists of changing the system structure. This implies the design and evaluation of a variety of system architectures to select *at run-time* the ones with a better trade-off. The analysis/selection implies solving a trade-off between performance/quality/cost attributes. However, not all selected architectures can be realised, something that could contribute to mitigating run-time contingencies from a general perspective.

The same applies to behavioral self-adaptation. Behavior selection involves the execution of a new action plan, i.e., deciding which actions to execute and in which order to enact a suitable response to run-time contingencies. In both adaptation paradigms, the designer is relieved from the burden of predicting all possible input states and selecting the corresponding output actions.

Both structural and behavioral self-adaptation have been widely explored in the domain of adaptive systems. However, most of the approaches are *ad hoc* adaptation mechanisms devoid of the *deep knowledge* that systems engineering leverages when solving complex problems.

In this work, we reflect on the use of ontologies as a knowledge tool to capture a common understanding of entities and patterns in the domain of the system [3]. This knowledge combined with logical reasoning can be used to gain insight into the system operation at run-time and how the system shall change to overcome arbitrary events. Ontologies are used here to empower by generalisation the capability for self-adaptation. The system design knowledge is available *to the system itself* during the system operation to select reconfiguration actions in response to expected or unexpected input events, i.e., contingencies that may happen during the system operation.

We discuss a methodology to use ontologies in systems

autonomy through a *general systems framework* that enables the decoupling of the system design knowledge from the concrete application. This enables the provision of a reusable module for general system reconfiguration. This enhances reusability, modularity, and scalability in the conceptualization and realization of self-adapting agents.

As a concrete example, we show the use of this technology in the mobile robot domain. The systems engineering knowledge of the robot system is captured in an ontology that is then used at run-time to overcome contingencies. In this article, we use the experience and benefits of these applications to define future lines to address more complex missions and environments. This is a lessons-learned descriptive article. It remains pending to compare the benefits of our approach measured against other strategies to augment the autonomy and resilience of systems.

The paper is organized as follows: Sec. II summarises the state of the art concerning adaptation in knowledge-driven agents. Sec. III introduces the advantages of using ontologies and an explicit mission definition to generalise and augment the autonomy levels of systems. Sec. IV presents some usage scenarios and how a general ontology shall be particularized to create an operative knowledge base. Sec. V describes some desired features for the self-adaptation ontology after evaluating the usage scenarios. Sec. VI concludes the article.

II. Related work

Systems are conceived and designed to fulfill a mission. The system triggers a self-adaptation action when there is a deviation in its operating conditions, so it shall modify its structure or behavior to reach its fundamental objective.

The notion of self-adaptation is directly linked to the concept of *affordance*. The concept of affordance was introduced by Gibson. To him, an affordance is equally a fact of the environment and a fact of behavior; it is a relationship that points both ways, to the environment and to the observer [4]. In robotics, affordances are usually seen as a relationship between the environment and the agent [5]. The adaptation must be directed towards providing the system with affordances according to the run-time contingencies it may arise. In [6], a descriptive affordance theory is presented. They focus on inferring the affordances of objects in the reachable region of the environment through ontologies. The robot understands the capabilities of the perceived object and how to interact with it. The same concept of affordance is used in [5] to design a robot through an ontology design pattern.

[7] takes a further step and uses a knowledge base through graph queries, to create an action plan for industrial mobile manipulators. This plan is adapted in case of run-time failures or performance degradation. In our work, the reconfiguration is also directed towards creating an adapted plan that matches the run-time contingencies when they arise. This adapted plan might be a new sequence of actions to overcome the unexpected events, like the solution proposed in [7]. However, our approach also considers performing the same actions by means of other alternative components or algorithms to maintain the initial plan. In this case, we take advantage of both structural and functional redundancy to improve the mission fulfillment probability in case of contingencies.

To direct the self-adaptation towards the mission, we use reasoners over ontologies. Ontologies in robotics are a hot topic, as can be seen from the standardization activities being carried out at the IEEE[1]. Neto [8] evaluates if the concepts present in the Core Ontology for Robotics and Automation (CORA) can be mapped at run-time in terms of performing a complex mission. They map Robot Operating Systems (ROS) elements to CORA's classes in a search and rescue application. In our case, we also base our ontology on standard ontologies such as CORA to define a complex mission. However, we extend the ontology with system monitoring concepts, as we will use it to supervise how the operation is being executed and to measure the run-time performance of the system.

Self-monitorization is a key feature for autonomous systems. It allows closing a control loop over the performative ability. Similar to classical controllers which aim to maintain a certain state in the system through observation and control actions, we aim to fulfill a mission within some quality level. In this case, the control actions correspond to the system self-adaptation and monitorization.

In general, the purpose of our work is to bridge the engineering run-time gap, making the system knowledge flow to the autonomous system cognitive engine. This knowledge will be used first to monitor the robot operation. Then, in case of contingencies, self-adaptation will be used to mitigate the impact of such contingencies on the mission accomplishment.

III. Ontologies and mission definition

Computer ontologies are specifications of conceptualizations [9]. A conceptualization is an abstract, simple perspective of some area of interest. The formalised concepts are used to build a *knowledge base* through classes, individuals, axioms, and relationships among them. Ontologies define a shared vocabulary for a domain, allowing collaboration and reusability. In our case, we target a general ontology based on autonomous systems, defining concepts related to their capabilities and the mission they shall accomplish. This general ontology can then be particularized —domain and system-focused— for different autonomous systems to be usable at run-time.

Our solution uses a system architectural model to explicitly decouple the handling of common undesired events and contingencies from the specific normal actions a system may take when doing its activities. This perspective encourages reusability and facilitates the labor of system architects when adding self-adaptation to extant designs (or even to deployed systems). The main challenge is the creation of an ontology general enough to be applicable to a variety of systems but with a suitably specific part to be exploited at run-time. For this reason, the ontology provided is structured at three levels of abstraction, namely: i) general systems, ii) application,

[1]See for example IEEE CORA Standard and RAS working groups P1872.1 and P1872.2

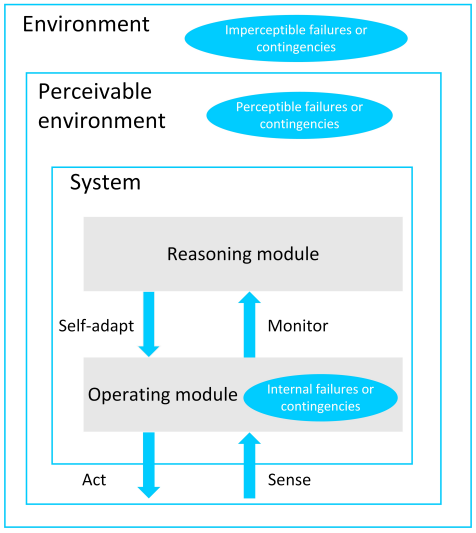


Fig. 1. System definition for self-adaptation and its information flows.

domain iii) specific application. Most of the rules that govern the self-adaptation are defined at the general system level, so the solution can be used for any component-based system. We also provide a domain-level containing a set of common elements to store the shared information within a given field. Lastly, the architect of the specific system must define a set of functional alternatives and link them to the components each of them require. With this structure, it is possible to use the ontology to reason in general system theory terms (i.e., objective, functions, components, etc.) and select, at run-time, the best design available among all concrete implementations defined by the system architect.

This ontology provides a baseline to metacontrol system action (see Figure 1). The reasoning module, which includes the knowledge base, can communicate with the operating module in charge of controlling the robot behaviour. The reasoning module monitors the system state and if the operating conditions have changed. Then, the ontology is used to reason about the perceived information, detect an anomaly, and infer that a reconfiguration action is in need. In this case, the knowledge base will provide the best available design, at the software or hardware level, to perform a self-adaptation in the operating module. Note that we have emphasized in the figure the internal failures and contingencies the robotic system may face as well as the perceptible and non-perceptible ones present in the environment.

In [10], the authors establish a research roadmap for self-adaptive software systems. For them, the most important features are: i) the design space regarding alternative solutions, ii) the processes and how they change, iii) the decentralization of control loops to reach system-wide quality goals, and iv) a practical run-time verification and validation. We use these guidelines as a starting point, and then we expand and redefine them in terms of requirements for the conceptualization of an ontology that drives self-adaptation in autonomous systems. Note that the heterogeneity present in general systems — hardware, software, humans, etc.— implies a higher degree of abstraction from what is needed for software-only systems.

The SE design and implementation of systems are always focused on solving a problem. In other words, an agent is designed to operate in a certain manner to complete a mission. This mission is divided into a set of measurable sub-objectives to evaluate the progress of the mission during the operation.

We use Quality Attributes (QA) to measure the quality level at which a certain system operation is performed. These QA correspond to different aspects of the system such as safety, energy consumption, performance, etc. It is important to note that the definition of a given QA strongly depends on the specific task the system is performing and the context in which the mission is performed.

A simple example is a patrolling robot. Some QAs associated with this mission are *safety*, as the robot speed and the distance to pedestrians cannot exceed a certain threshold, and *energy consumption*, as it cannot fully drain the battery, and *performance*, as the patrolling shall cover some area in a defined time frame.

This mission has several design alternatives such as the algorithm used to cover an area, the obstacle avoidance strategy, sensors used for navigation, etc. These design alternatives can have different QA estimations. The reasoner can use such estimators to select the best design alternative to perform self-adaptation. For example, if the robot is patrolling and enters a room full of mirrors, the laser used to perceive the environment will not be reliable anymore. The reasoner must detect that the safety of the mission is compromised and trigger a reconfiguration to use another sensor, such as an RGBD camera to maintain the operation. This sensor may have lower performance and safety QA estimated than the previous architecture but allows the mission accomplishment within a compromise between all QAs.

Therefore, besides the mission definition, a critical aspect is the architecture of the system itself. How the system is built in terms of which are its sensors, actuators, and its structural and functional components and which capabilities they provide. The knowledge base shall be capable to include all engineering knowledge on these components, such as in which conditions they are usable and a measure of their trustworthiness. The software architecture must be included in the ontology as well. This includes the processes required to use a certain module and the potential collisions with other modules. The knowledge base shall include what the functionality a module provides and which inputs and outputs it provides. In the previous example, the localization software module can use a Kalman Filter or the Adaptive Monte Carlo Localization (AMCL) module. Both algorithms provide as output a location of the robot and an estimation of its accuracy. They differ on the memory required to process the sensor data and the input sensor information used.

Lastly, context plays a key role in the design of self-adaptive systems. For instance, the estimation of accuracy in the location provided by these algorithms is subject to the

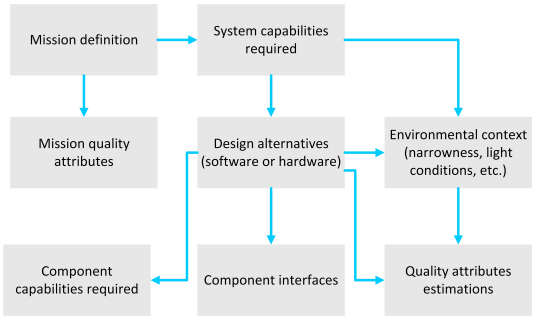


Fig. 2. Concepts for mission assurance in self-adapting autonomous systems.

reliability of the sensory inputs which depends on the environmental features. For autonomous systems operation, the context is especially important when understanding the quality with which a system performs a mission. For example, a certain robot controller might perform better in terms of safety in wide environments, while having low safety performance in narrow environments. This requires a set of diagnostic tools that can measure the QAs as well as to detect and inform about any contingency or component failure that may appear. In other words, it should be able to trace back the source of a given error (i.e., which component has failed, what quality attribute is not being met, or what changes in the environment have occurred.). Figure 2 depicts the concepts discussed above.

IV. Usage scenarios

Our group has been previously working on the conceptualization of ontologies for autonomous and self-adapting systems [11]. In [12], TOMASys (Teleological and Ontological Model of an Autonomous System), a general meta-model to augment the autonomy level of systems is defined. This ontology addresses the basic aspects depicted in Figure 2.

This autonomous system metamodel has been transformed into an ontology and used in different systems, evaluating the generality of the ontology and to what extent it is feasible to exploit it at run-time. A formal definition of the ontological model used can be found in [13].

In the recently finished EU-funded Metacontrol for Robot Operating System (MROS) project[2], we have addressed a patrolling corridor mission with a TurtleBot2 robot. We use an ontological reasoner to face a simulated laser error at a random instant. These data were corrupted by generating realistic data coherent with something blocking the laser, or with a laser misalignment, maybe due to a hit or fall of the robot. When a failure in the sensor was detected, the system reconfigured itself to use an RGBD camera and reduce its speed to not compromise the integrity of the robot and other elements in the surroundings, as the update rate of camera sensory inputs is lower than in the laser sensor. Note that the functional reasoning happens at the abstract level of needs and capabilities and what can provide them —as an engineer would do— and not at the lower level of concrete realizations.

[2]https://robmosys.eu/mros/

The main experiment carried out measures the recovery time for a laser failure using ontology-driven reconfiguration. After 50 iterations of the experiment, the time required is 1.995 s with a standard deviation of 0.478. Without it, the estimated recovery time for this failure is about 300 s (indeed tied to system maintenance involving humans) [14].

Another application in which we have evaluated the ontology-driven self-adaptation is the UX-1 autonomous underwater robot. The UX-1's mission is to explore and map underground flooded mines. It shall provide geological, visual, and spatial data of hazardous areas that currently are expensive and risky to explore.

This ontological approach was used to deal with thruster failure [13]. A naive way of handling this is disabling the paired symmetric thruster. However, we used a knowledge base to recalculate the force allocation in properly working thrusters and maintain its functioning. The experiments run and described in the article show that the mission is accomplished in 203.2 s with the naive approach, i.e., disabling the symmetric thruster in case of failure. In the case of using the ontology to reallocate the forces and maintain the usage of all functioning thrusters, the mission is completed in 168.7 s.

The TOMASys ontology is built upon a general meta-model based on design patterns. Its concepts have been proven feasible at run-time for reconfiguration in navigating applications in complex environments such as patrolling, in corridors, or in flooded mines. This ontology serves as a start point to design the self-adaption ontology, a general knowledge-based designed to track complex mission definitions and capable to capture the variability of status in autonomous systems. In [14], we focused on describing what shareable ontologies could do to streamline the autonomous robot development and exploitation, summarizing the state of the art, and presenting some test beds. In this article, we employ that experience to establish a roadmap to tackle more complex missions and environments with this approach.

V. Lessons learned

In the two usage scenarios described above, we evaluate the use of the TOMASys meta-model as an ontology to drive different robots' adaptation towards accomplishing a mission. In those test beds, the mission was quite simple, as it only targeted navigation, even though the environmental conditions were undoubtedly complex and harsh. The TOMASys meta-model is focused on the functional capabilities and interfaces between the structural components that compose the system [12], concepts that go beyond mere robot navigation.

Here we discuss some critical aspects we have encountered to reach a complete ontology that supports general self-adaptation. We need to focus on: i) how a mission is decomposed and how the processes running to ensure a correct operation are managed together, ii) overcome some limitations in the ontology implementation, and iii)the importance of modeling the environment through contextual information to understand what is happening in its surroundings at run-time.

In this direction, we are currently working on a reconfigurable miner robot to perform selective mining in the ROBOMINERS EU-funded project[3]. This robot may display a higher level of resilience and adaptation because it shall operate in harsh conditions —e.g., flooded or deep mines.

A. Complexity of mission and processes handling

The TOMASys implementation in real robots helped identify two limitations: (i) limiting the concreteness of the objectives handled and (ii) the management of other tasks and processes running in the system.

The first issue relates to the nested complexity of real-world missions. The solution shown here can only handle one objective at a time and has no support for low-level tasks with nested objectives. However, it is common that missions are composed of complex orchestrated tasks. Our general ontology framework must support the conceptualization of subtasks and how they contribute to the main objective. Therefore, we must extend our work to handle multiple goals within a hierarchy. Endowed with this capability, the system adaptation may fully support real-world ambitious structured missions. For example, the ROBOMINERS mission we shall accomplish includes recognizing a flooded mine and its minerals, navigating through the mine, interacting with the mine to selectively extract the mineral, and reduce the grain size to transport the material, while at the same time the robot shall keep itself operational, able to return to the base, and cooperative with a human operator to the surface.

Besides the complex goal structure, some aspects of the robot activity are soft and hard real-time and the associated reasoning processes are necessarily time-dependent. The knowledge base must gain insight into handling constraints imposed by time. In the design phase of physical systems, time is often analyzed using logical clocks based on an order relation between events (e.g. the *happened-before* relation). However, when the system is physically deployed and multiple sub-systems are coordinated, the quantified execution times needed by each process and the potential priority-based dependencies among them shall be addressed.

This problem is quite noticeable in self-adapting physical systems, as reconfiguration takes a certain amount of time, but the physics of the system elements —or the mine— does not stop. The system architect needs to decide on what happens in the temporal window since the system is in error until it has been re-instated to keep its operation. Moreover, the reconfiguration must be time-bounded for safety reasons, in case the adaptation fails and the system cannot reach the desired state. For an improved implementation of self-adapting mechanisms driven by knowledge, the ontological framework used needs to take into consideration the temporal restrictions of other subsystems and the time consumed by the reconfiguration.

Addressing time in ontologies is not new, however, and a future ontology for autonomy could leverage other's efforts. Time has been addressed from an ontological perspective in the Suggested Upper Merged Ontology (SUMO) [15] and in the Time Ontology in OWL [16]. Besides pure ontological work, the OMG Modeling and Analysis of Real-time and Embedded systems (MARTE) specifications address the problem of time management as a core system aspect -both hardware and software- for model-driven real-time systems development [17].

B. Type of logic for the self-adaption ontology

The TOMASys-derived ontology has been encoded in OWL2, a language for ontology formalization based on description logics [18]. The underlying description logic underlying OWL2, SROIQ(D), is simple enough and effective for database querying and the semantic web, but in the systems world, there are some issues to address.

The logic used in OWL and its reasoner is monotonic. This means that when a formula is added, the set of consequences is never reduced. Controller implementations are based on repetitive perceptions about the system state and the status of its environment, so the contents of the knowledge base shall be dynamic, at least in part. This dynamicity cannot be easily handled by the type of logic and reasoner that we are using. In the current solution, we overcome this problem by extending the whole system with revision operations through a programmable API: removing previous statements from the database and asserting the valid ones at run-time. However, this practice is quite time-consuming and may indeed threaten the integrity of the ontology.

To truly overcome these limitations, other types of logic must be evaluated. One option is the use of a *modal logic* that implements dynamic logic according to a possible world's semantics. *Temporal description logic* is another option because the system handles time-dependent processes and its sub-systems are bounded to different life cycles. Using the same kind of resources that we used here, Knorr, Hitzler, and Maier [19] proposed some extensions to OWL and its associated rule languages to unify logics in the semantic web.

Lastly, when dealing with real-world vague concepts and imprecision, *fuzzy logic* extensions may be employed in combination with other types of logic, especially during the adaptation processes in which the system is not fully operational.

C. Use of contextual information

While within this work we investigated the modeling of the contextual information related to the system as a whole, we still need to do further exploration in terms of tracing the contextual information and linking it with corresponding system components or subsystems. We consider that the context relates to the functioning of the components (e.g., using a camera in the localization subsystem), the functioning of a subsystem (e.g., using the navigation subsystem with erroneous localization), or to the functioning of the whole system (e.g., a robot navigates in a dark environment). These levels of abstraction require a clear separation of concerns between the context and the hierarchical model of the system, exploring

[3]https://robominers.eu/

the dependencies between the contextual information, the system, its components, and the quality to which a mission is executed. To reason effectively and provide assurance on the behavior of self-adaptive systems at run-time, it is required that researchers make additional efforts towards context modeling for ontological solutions. This means exploring how effectively different approaches model the dynamicity of the operational environment and how well they capture the uncertainty of the perceived context.

Also note that the fault detection mechanism previously implemented in our system considers just the case of the component being completely out of service. However, fault detection could be enhanced by detecting intermediate states, misreadings, or impossible values from sensors, which can result in fully unavailable components or in merely degraded system performance. Measuring the quality of the information coming from a given sensor can help in designing solutions that can handle the system QA in a graded way and improve the capacity of the system to better pursue a given goal. For the ROBOMINERS miner robot, a degraded but still operational movement capability is critical for survival (e.g., to escape a collapse or to find a robot leg spare).

Another research direction is to causally investigate the origin of a fault, as a given error may be due to a component failure, due to changes in the environment, due to external inputs, or even due to emergent phenomena that are difficult to trigger using a localized measurement approach. Researchers need to do further investigation on how to link these contingencies to specific changes in the QA values to improve the performance of the overall system.

VI. Conclusions

Technical systems are designed to produce a capability in answer to a need. Here we describe an ontology-driven architecture for self-adapting systems that allows the exploitation of explicit engineering knowledge about such capability-need relations. The work described is part of the EU-funded MROS and ROBOMINERS projects. The last one addresses these issues for an autonomous selective mining robot operating in harsh conditions such as deep or flooded mines. In this paper, we have summarized our experiences with ontologies for less complex missions and propose new research directions in using an ontology for resilient, self-adaptive systems. These lesson learned experiences let us identify some interesting future work.

The ontological solution aims to fuel the mission-targeting meta-control loop from a system-level knowledge perspective. The run-time reasoner uses the information coming from a perception system focused on detecting undesirable changes both in the environment and in the system itself. This perception system injects state knowledge into the module in charge of the system diagnostics. Further development of this subsystem to provide the ontology with causal, time-aware, accurate, and detailed information can significantly improve the whole self-adapting structure.

Acknowledgment

This work was supported by the ROBOMINERS (Grant Agreement No. 820971) project and by the RobMoSys-ITP-MROS (Grant Agreement No. 732410) project with funding from the European Union's Horizon 2020 research and innovation programme and by the *Programa Propio* grant supported by the Universidad Politécnica de Madrid.

References

[1] H. A. Simon, *The Sciences of the Artificial*. Cambridge, MA: {MIT} Press, third ed., 1996.

[2] D. D. Walden, G. J. Roedler, K. J. Forsberg, R. D. Hamelin, and T. M. Shortell, "Systems engineering handbook. a guide for system life cycle processes and activities v4.0," Technical Publication INCOSE-TP-2003-002-04, international Council on Systems engineering (INCOSE), Hoboken, NJ, USA, 2015.

[3] R. Sanz, J. Bermejo, J. Morago, and C. Hernández, "Ontologies as Backbone of Cognitive Systems Engineering," in *Proceedings of AISB CAOS 2017: Cognition And OntologieS*, (Bath, UK), 2017.

[4] J. J. Gibson, *The ecological approach to visual perception*, p. 332. Boston, MA, USA: Houghton Mifflin, 07 1979.

[5] L. Asprino, A. Nuzzolese, A. Russo, A. Gangemi, V. Presutti, and S. Nolfi, *An ontology design pattern for supporting behaviour arbitration in cognitive agents*, pp. 85–95. 12 2017.

[6] D. Beßler, R. Porzel, M. Pomarlan, M. Beetz, R. Malaka, and J. Bateman, "A formal model of affordances for flexible robotic task execution," in *Frontiers in Artificial Intelligence and Applications*, vol. 325, pp. 2425–2432, IOS Press BV, aug 2020.

[7] A. Kattepur and B. P, "Roboplanner: Autonomous robotic action planning via knowledge graph queries," in *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*, (New York, NY, USA), p. 953–956, Association for Computing Machinery, 2019.

[8] A. B. d. O. Neto, J. A. Silva, and M. E. Barreto, "Prototyping and validating the cora ontology: Case study on a simulated reconnaissance mission," in *Latin American Robotics Symposium (LARS*, pp. 341–345, 2019.

[9] T. R. Gruber, "A translation approach to portable ontologies," *Knowledge Acquisition*, vol. 5, no. 2, pp. 199–220, 1993.

[10] R. de Lemos *et al.*, *Software Engineering for Self-Adaptive Systems: A Second Research Roadmap*, pp. 1–32. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013.

[11] J. Bermejo-Alonso, C. Hernandez, and R. Sanz, "Model-based engineering of autonomous systems using ontologies and metamodels," in *ISSE 2016 - 2016 International Symposium on Systems Engineering - Proceedings Papers*, 2016.

[12] C. Hernández, *Model-based Self-awareness Patterns for Autonomy*. PhD thesis, Escuela Técnica Superior de Ingenieros Industriales - Universidad Politécnica de Madrid, 2013.

[13] E. Aguado, Z. Milosevic, C. Hernández, R. Sanz, M. Garzon, D. Bozhinoski, and C. Rossi, "Functional self-awareness and metacontrol for underwater robot autonomy," *Sensors*, vol. 21, 2021.

[14] E. Aguado and R. Sanz, "Using Ontologies in Autonomous Robots Engineering," in *Robotics Software Design and Engineering [Working Title]*, IntechOpen, apr 2021.

[15] I. Niles and A. Pease, "Towards a standard upper ontology," in *Proceedings of the International Conference on Formal Ontology in Information Systems*, (New York, NY, USA), p. 2–9, Association for Computing Machinery, 2001.

[16] J. R. Hobbs and F. Pan, "An ontology of time for the semantic web," *ACM Transactions on Asian Language Information Processing*, vol. 3, p. 66–85, Mar. 2004.

[17] A. Louati, K. Barkaoui, and C. Jerad, "Time properties verification of UML/MARTE real-time systems," in *Proceedings of the 2014 IEEE 15th International Conference on Information Reuse and Integration*, pp. 386–393, 2014.

[18] F. Baader, I. Horrocks, C. Lutz, and U. Sattler, *An Introduction to Description Logic*. Cambridge University Press, 2017.

[19] M. Knorr, P. Hitzler, and F. Maier, "Reconciling OWL and non-monotonic rules for the semantic web," in *20th European Conference on Artificial Intelligence.*, vol. 242, pp. 474–479, IOS Press, 2012.

Analysing Ontological Requirements: A Journey from Requirements to Code and Back

Alba Fernández-Izquierdo
Ontology Engineering Group
Universidad Politécnica de Madrid
Spain
albafernandez@fi.upm.es

María Poveda-Villalón
Ontology Engineering Group
Universidad Politécnica de Madrid
Spain
mpoveda@fi.upm.es

Raúl García-Castro
Ontology Engineering Group
Universidad Politécnica de Madrid
Spain
rgarcia@fi.upm.es

***Abstract*—Ontological requirements play a key role in ontology development as they determine the knowledge that needs to be modelled. The specification of these requirements, as well as its translation into tests, is not a trivial task. Due to the fact that there is not a standardised way to specify ontological requirements, and that sometimes they are vague or ambiguous, their translation into code or tests is not always direct, so that the same requirement can result in different structures in the ontology. In this work an empirical analysis has been carried out in order to study how ontology engineers define requirements and their impact in the actual ontology. More precisely, the difference between the requirements definition and the ontology implementation has been analysed.**

***Index Terms*—Ontology requirements, Ontology development, Ontology verification**

I. Introduction

In recent years, the definition of ontological requirements (e.g. [1], [2]) and its automatic formalization into axioms or tests (e.g. [3], [4], [5]) have been studied with the aim of reducing the time consumed by ontology developers during the ontology verification activity.

However, an accurate definition of ontological requirements is not a trivial task and, therefore, neither is its automatic translation into a formal language. Due to the fact that some requirements are ambiguous [6] or vague, its transformation into axioms or tests is not usually direct and, consequently, it is very difficult to automate such translation. In addition to this, Dennis et al. [3] have analysed that some ontological requirements include *presuppositions*, which is a proposition whose truth is a precondition to assessing the truth or falsity of a sentence. As an example, the requirement "*Which pizzas contain chocolate?*" implies that there are pizzas with chocolate and pizzas without chocolate, even though it is not explicitly indicated in the requirement. Accordingly, to translate requirements into axioms or tests in order to efficiently verify ontologies, the implicit knowledge of these presuppositions needs to be considered and included.

After exploring this situation, and with the long-term goal of automating as much as possible the translation of requirements into axioms or tests, the work presented here aims at analysing how ontology engineers are currently defining ontological requirements, and the impact of such requirements in the consequent ontology implementation. The motivation of this work is to identify the limitations of the ontological requirements specification in order to avoid ambiguities, make the requirements more precise, and in the long term being able to automate testing. To do so, the following research question was brought up:

- *RQ1. What is the difference between what is explicitly defined in requirements and the consequent ontology implementation?*

The analysis proposed in this work also took into account that ontological requirements can be defined in two different environments which have different goals, i.e., to support a particular project and to provide a standard model[1] to some community.

In order to analyse whether the goal of the ontology influences the requirements definition, the following research question was also introduced:

- *RQ2. Is the definition of requirements in non-standard ontologies different from the definition of requirements in standard ontologies?*

After collecting a set of requirements from existing corpora of ontology requirements, an empirical analysis to address these two research questions was proposed.

In this analysis, the expected OWL constructs and expressivity of the requirements of different standard and non-standard ontologies were compared with their corresponding ontology implementation. It is worth mentioning that the work presented here is focused on the difference between what is defined in the ontological requirements and what is modelled in the ontology implementation. Therefore, the analysis about how the ontologies are built and the transformation from requirements to each ontology element are out of scope.

The paper is organised as follows. Section 2 presents the lexico-syntactic patterns used in the study to analyse ontological requirements in a systematic way. Section 3 describes the process we followed and section 4 describes the obtained results. Section 5 presents the discussion of the obtained results and Section 6 describes the related work on ontological requirements analysis. Finally, Section 6 presents the conclusions we obtained and gives an overview on future work.

[1] In this work standard ontologies are considered as those that are endorsed by standardisation initiatives such as W3C, ETSI or oneM2M.

II. Lexico-Syntactic Patterns and the CORAL corpus

Lexico-syntactic ontology design patterns are understood as "formalised linguistic schemas or constructions derived from regular expressions in natural language that consist of certain linguistic and paralinguistic elements, following a specific syntactic order, and that permit to extract some conclusions about the meaning they express" [7]. Daga et al. [8] and Montiel-Ponsoda [6] proposed a corpus of 17 lexico-syntactic patterns (LSPs)[2], which includes patterns for subclassOf relations, object property relations or part-whole relations. Each lexico-syntactic pattern includes: a) identifier, b) formalisation and c) examples of sentences in natural language which match with such pattern.

It is worth underlying that there can be several formalisations for a lexico-syntactic pattern. As an example, the pattern LSP-SC-EN in Table I has associated the formalisations[3]:

1) "[(NP<subclass>,)* and] NP<subclass>be [CN-CATV] NP<superclass>"
2) "*There are QUAN CN-CATV NP<superclass> PARA [(NP < subclass> ,)* and] NP<subclass>*"
3) "*[A(n) | QUAN] example of | CN-CATV NP<superclass> be | include [PARA] [(NP<subclass>,)* and] NP<subclass>*"

A later work is CORAL [9], which extends this existing corpus with 12 new LSPs. In total, CORAL provides a dictionary of 29 LSPs (17 from [6] + the 12 new LSPs) and a corpus of 834 ontological requirements annotated with such LSPs. Both the dictionary of patterns and the annotated corpus are provided in the machine-readable format RDF, in CSV and in HTML as Zenodo[4] and Datahub[5] resources.

Both corpora were organised according to the types of correspondences between the lexico-syntactic patterns and the ontology design patterns (ODPs) [10] needed to implement them in an ontology. This categorisation can be used to identify whether the lexico-syntactic patterns are polysemous, i.e., the same LSP can result in several possible ontology implementations:

- **1 to 1 correspondence.** One lexico-syntactic pattern corresponds to one ontology design pattern. An example of this type of pattern is shown in Table I, which describes *subclassOf* relations.
- **1 to N correspondence.** One lexico-syntactic pattern corresponds to a combination of two or more ontology design patterns to be modelled in the ontology. An example of this type of pattern is shown in Table II, which describes *defined classes and subclassOf relations*. The LSPs that have more than 1 correspondence are considered polysemous LSPs.

[2]http://ontologydesignpatterns.org/wiki/Submissions:LexicoSyntacticODPs

[3]In the example: *QUAN* refers to quantifiers, *CN-CATV* refers to class names conveying classification, *NP* refers to noun phrases and *PARA* refers to paralinguistic symbols.

[4]https://zenodo.org/record/4432491

[5]https://datahub.io/albaizq13/coralcorpus

TABLE I
"1 TO 1 CORRESPONDENCE" LEXICO-SYNTACTIC PATTERN FOR SUBCLASSOF RELATIONS.

LSP Identifier		LSP-SC-EN
Formalisation	1	[(NP<subclass>,)* and] NP<subclass>be [CN-CATV] NP<superclass>
	2	There are QUAN CN-CATV NP<superclass> PARA [(NP<subclass>,)* and] NP<subclass>
	3	[A(n) | QUAN] example of — CN-CATV NP<superclass> be —include [PARA] [(NP<subclass>,)* and] NP<subclass>
Example	1	An orphan drug is a type of drug
	2	There are several kinds of memory: fast, expensive, short term memory and long-term memory
	3	Types of criteria for assessing applications are: quality, safety

TABLE II
"1 TO N CORRESPONDENCE" LEXICO-SYNTACTIC PATTERN CORRESPONDING TO DEFINED CLASSES AND SUBCLASSOF RELATIONS.

LSP Identifier		LSP-DC-SC-EN
Formalization	1	[A | any] NP<subclass> be [a | any] NP<superclass> REPRO VB [(NP<class>,)* and] NP<class>
	2	[A | any] NP<subclass> be [a | any] NP<superclass> PREP [(NP<class>,)* and | or NP<class
	3	[A | any] NP<subclass> REPRO VB [(NP<class>,)* and— or NP<class>be [a] NP<superclass>
	4	[A | any] NP<subclass> PREP [(NP<class>,)* and] NP<class> be VB [a] NP<superclass>
Example	1	A device is any machine or component that attaches to a computer
	2	A vegetarian pizza is a pizza without fish or meat
	3	A workflow that contains at least one business task is a business plan
	4	Animal with backbones are called vertebrates

III. Empirical setup

Taking into account the lexico-syntactic patterns described in Section II, a set of steps were followed in order to carry out the analysis. These steps, as well as their inputs and outputs, are summarised in Figure 1:

1) *Select input.* The input of the analysis was identified. This selection can be divided into three sub-steps:
   a) *Search ontologies.* To conduct the analysis between the requirements and the corresponding ontologies and to address the research questions presented in Section I, it was decided to carry out the analysis over the same number of standard and non-standard ontologies, as well as ontologies built by different authors, and of different topics. Accordingly, a set of 10 ontologies was selected. From these 10 ontologies, 5 were standard ontologies, and the remaining 5 were non-standard ontologies. The names and URIs of the collected ontologies are summarised in Table III. These 10 ontologies were associated to 767 different functional requirements.
   b) *Search lexico-syntactic patterns.* A set of lexico-syntactic patterns was identified, as described in Section II, in order to be able to systematically analyse the ontological requirements.
2) *Associate ontological requirements from collected ontolo-*

TABLE III
COLLECTED ONTOLOGIES WITH THEIR ASSOCIATED URIS

	Ontology name	Ontology URI	#R
Standard	SAREF	https://w3id.org/saref#	71
	SAREF4BLDG	https://w3id.org/def/saref4bldg#	98
	SAREF4ENVI	https://w3id.org/def/saref4envi#	58
	oneM2M	https://git.onem2m.org/MAS/BaseOntology/raw/3_6_0/base_ontology.owl#	58
	ODRL	https://www.w3.org/ns/odrl/2/	15
Non-standard	VICINITY Core	http://iot.linkeddata.es/def/core#	127
	Software Ontology	http://theswo.sourceforge.net/	90
	Video Game	http://purl.org/net/VideoGameOntology#	66
	Demc@re	http://www.demcare.eu/results/ontologies#	107
	BTN100	https://datos.ign.es/def/btn100#	77

gies with lexico-syntactic patterns. Each requirement was associated to its corresponding lexico-syntactic pattern in the CORAL corpus. Consequently, the expected OWL constructs needed to implement each requirement and their expressivity were obtained.

3) *Compare the expressivity of the requirements with the expressivity of the ontologies*. Finally, the OWL constructs associated to the ontology requirements were compared with the ontologies implementation. More specifically, the expressivity and the expected OWL constructs of the set of requirements were compared with the expressivity and actual OWL constructs of the collected ontologies. This comparison allows to identify whether there are OWL constructs added to the ontologies that are not deduced from the requirements. The goal of this analysis is to obtain: (1) what are the differences between the OWL constructs and expressivity in the requirements and in the ontologies implementation, and (2) what are the differences between the requirements of the standard ontologies and the requirements of the non-standard ontologies. The expressivity comparison considering each requirement individually is out of scope of the paper. However, the expected expressivity for each requirement can be consulted in the CORAL corpus.

IV. RESULTS

The results obtained from the comparison of the set of requirements and the implementations associated to the 5 selected standard ontologies and the 5 non-standard ones are presented in Table IV, which shows the OWL constructs for the selected standard ontologies, and in Table V, which displays the OWL constructs for the selected non-standard ontologies. In these tables, for each ontology (columns) it is indicated whether the requirements or the ontology implementation contain an OWL construct (rows) by means of a symbol in the cell. More precisely if the construct is expected according to the requirements and it appears in the ontology, then the symbol "✓" is included in both columns. If the construct is expected according to the requirements but it does not appear in the ontology the symbol "—" is included in the requirements column, leaving empty the code one. Finally, if the construct is not expected according to the requirements but it appears in the ontology, then the ontology code is marked with a "+" and the requirements column is empty.. For example, for the ODRL ontology in Table IV we can observe that the construct "SubClassOf" is expected to appear in the ontology according to the set of requirements (R column) and it does appear in the ontology implementation (O column) while the "Some Values From" is expected to appear but it does not, and finally the "Range" construct is not expected to appear but it does appear in the ontology code. In addition to the OWL constructs, the tables also include the expressivity associated to the set of requirements and the implementation for each ontology (last row). Axioms related to data are not shown in the tables, e.g., *allDifferent* or *sameAs*, due to the fact that they do not appear neither in the requirements nor in the ontologies implementation, and this analysis was focused on the TBox of the ontologies. The results provided in Tables IV and V represent the aggregated expressivity of the set of requirements associated with each ontology.

V. DISCUSSION

With the results of the analysis presented in the previous section the two research questions that were introduced in Section I can be addressed. It should also be taken into account that this work has analysed a subset of the ontologies published in the Web; i.e., only those that have openly available functional requirements.

A. RQ1. What is the difference between what is explicitly defined in the requirements and the consequent ontology implementation?

Regarding the first research question, Tables IV and V illustrate that the requirements are only related to a small set of the OWL constructs. In fact, the rate of related OWL constructs ranged from 15-35% of the total. Nevertheless, in the case of the associated ontology implementations they ranged from 38-69% of the total of OWL constructs. This difference in the percentage of related OWL constructs shows that the requirements are very general in comparison with what is actually modelled in the ontology. Therefore, part of the ontology development depends on the ontology engineers. Moreover, the requirements use to be related always with the same OWL constructs:

* Class
* Subclass of
* Property
* Some values from

However, there are a set of OWL constructs that are never associated to the requirements, but they are common in ontology implementation:

* Disjoint class
* Transitive property
* Functional property
* Equivalent class
* Subproperty of

TABLE IV
OWL CONSTRUCTS ASSOCIATED TO EACH STANDARD ONTOLOGY (R REPRESENTS WHETHER THE CONSTRUCT IS EXPECTED TO APPEAR ACCORDING TO THE REQUIREMENTS AND O REPRESENTS WHETHER THE CONSTRUCT APPEARS IN THE IMPLEMENTATION)

	Ontologies endorsed by standardisation initiatives									
	SAREF		**ODRL**		**SAREF4BLDG**		**oneM2M**		**SAREF4ENVI**	
OWL constructs	**R**	**O**	**R**	**O**	**R**	**O**	**R**	**O**	**R**	**O**
Class	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
SubClassOf	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Property	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
SubPropertyOf		+						+		
Domain		+						+		
Range		+		+		+		+		
Individual	✓	✓								+
Equivalent Class		+								
Equivalent Property										
All Values From		+		+		+		+	—	
Some Values From	✓	✓	—		—		✓	✓	✓	✓
Min Cardinality	✓	✓					✓	✓		
Max Cardinality		✓			✓	✓		✓	✓	✓
Cardinality		✓	✓	✓		✓	✓	✓	✓	✓
Intersection Of										
Datatypes	✓	✓	—		—		✓	✓	✓	✓
One Of								+		
Disjoint With	✓	✓		+		+		+		+
Union Of	✓	✓		+		+		+		
Complement Of										
Has Value		+				+				
Inverse Of		+		+		✓	✓			+
Transitive Property		+		+		+				+
Symmetric Property										+
Functional Property		+		+		+				+
Inverse Functional Property										+
Expressivity	$\mathcal{ALCQ}^{(\mathcal{D})}$	$\mathcal{SROIQ}^{(\mathcal{D})}$	$\mathcal{ALEQ}^{(\mathcal{D})}$	$\mathcal{SHO}^{(\mathcal{D})}$	$\mathcal{ALEQ}^{(\mathcal{D})}$	$\mathcal{SIQ}^{(\mathcal{D})}$	$\mathcal{ALEQ}^{(\mathcal{D})}$	$\mathcal{ALCHQ}^{(\mathcal{D})}$	$\mathcal{ALEQ}^{(\mathcal{D})}$	$\mathcal{SOIQ}^{(\mathcal{D})}$

✓: expected in requirements and appears in ontology
+ : not expected in requirements but appears in ontology
—:expected in requirements but missing in ontology

These tables point out that this set of OWL constructs, which is never directly deduced from the requirements, is added to the ontology always as a consequence of modelling decisions of ontology engineers.

B. RQ2. Is the definition of requirements in non standard ontologies different from the definition of requirements in standard ontologies?

Regarding the second research question, Table IV was also compared with Table V. From the comparison, it could be determined that the requirements in standard ontologies include more OWL constructs than requirements in non-standard ontologies. In the case of the standard ontologies, the requirements of the major part of the ontologies range from 23-35% of the total of OWL constructs. However, in the case of the non-standard ontologies, the requirements range from 15-35% of the OWL constructs. These requirements of non-standard ontologies mainly range from 15-19%, except for the Core ontology, whose requirements include more OWL constructs.

This fact suggests that during the requirements specification process of standard ontologies the requirements were more detailed that in the case of non-standard ontologies. In fact, it seems that the requirements of non-standard ontologies are defined at a very general level owing to the fact that the expressivity is mainly limited to $ALE^{(D)}$. Hence, these requirements are limited to the representation of attributive language and full existential qualification. However, the actual ontologies have much higher expressivity, including characteristics such as cardinalities or inverse relationships. This situation can be detected in the Core ontology, where there are several OWL constructs, such as complementOf, that are not considered in the requirements but are contained in the ontology. This could indicate that the requirements are not updated with the needs that arise during the ontology development.

After the analysis of these results it can be concluded that, even though there are slight differences between the requirements of standard and non-standard ontologies, in both situations the requirements are related to a small set of OWL constructs, being the difference between the requirements and ontology implementations high in both cases.

It is true that the requirements should be defined together with domain experts, which do not need to have knowledge about ontologies. However, after analysing the collected requirements, and the considerable differences between what is defined in the requirements and what is implemented in the ontologies, the results showed that it should be necessary to improve the *ontology elicitation activity* [11] in order to produce more precise requirements. During this activity, the ontology engineer should identify ambiguous expressions, such as the ones presented in the lexico-syntactic patterns described in this work, and replace them with more specific terms. Ontology engineers should interview domain experts

TABLE V
OWL CONSTRUCTS ASSOCIATED TO EACH NON-STANDARD ONTOLOGY (R REPRESENTS WHETHER THE CONSTRUCT IS EXPECTED TO APPEAR ACCORDING TO THE REQUIREMENTS AND O REPRESENTS WHETHER THE CONSTRUCT APPEARS IN THE IMPLEMENTATION)

	Non-standard ontologies									
	VICINITY Core		**SWO**		**Video Game**		**Demc@re**		**BTN100**	
OWL constructs	**R**	**O**	**R**	**O**	**R**	**O**	**R**	**O**	**R**	**O**
Class	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
subClass Of	✓	✓	✓	✓	✓	✓		+	✓	✓
Property	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
subProperty Of		+		+		+		+		+
domain		+		+		+		+		+
range		+		+		+		+		+
individual		+	✓	✓	—		✓	✓	—	
Equivalent Class				+				+		
Equivalent Property										
all Values From		+						+		+
some Values From	✓	✓					✓	✓	—	
min Cardinality	—									
max Cardinality		+								+
cardinality	—			✓				+		+
intersection Of	✓	✓		✓						
datatypes	✓	✓	✓	✓		+	✓	✓	✓	✓
oneOf								+		
disjointWith		+		+				+		
unionOf		+		+		+		+		+
complement Of		+							—	
has Value										
inverse Of		+		+		+		+		
Transitive Property		+		+				+		
Symmetric Property	✓	✓		+		+		+		
Functional Property		+								
Inverse Functional Property		+								
Expressivity	$ALEIQ^{(D)}$	$SRIQ^{(D)}$	$ALE^{(D)}$	$\boldsymbol{SHOIQ^{(D)}}$	$ALE^{(D)}$	$ALUHI^{(D)}$	$ALE^{(D)}$	$SHOIQ^{(D)}$	$ALE^{(D)}$	$SHIQ(D)$

✓: expected in requirements and appears in ontology
+ : not expected in requirements but appears in ontology
—:expected in requirements but missing in ontology

so that they could, in a transparent way to these domain experts, ask the appropriate questions to make more accurate the requirements specification.

By refining this elicitation activity, which is depicted in Figure 1, the difference between the requirements and the ontology implementation may be reduced, providing a more precise outlook about what needs to be modelled in the ontology. Moreover, the ontology engineers could reduce the ambiguity of the requirements, easing the automatic translation into axioms or tests.

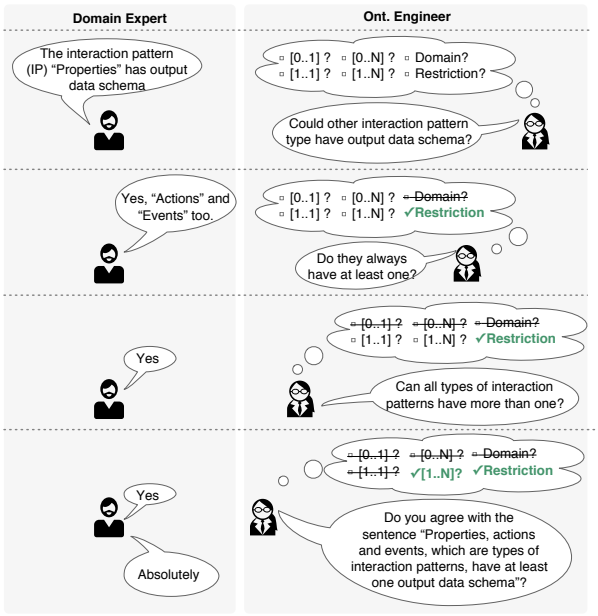


Fig. 1. Example of the elicitation activity in ontology development.

VI. RELATED WORK

Studying ontological requirements in the ontology development process is not a new idea itself [12] [11]. There are several works related mostly to the analysis of lexico-syntactic patterns from ontological requirements, while other works are focused on how the requirements can be translated into SPARQL queries. The main goal of all these works is to ease the automation of the requirements formalization process.

Regarding the analysis of lexico-syntactic patterns, several works have analysed how the requirements are constructed. Daga et al. [8] provides a set of lexico-syntactic patterns extracted from ontological requirements which have a direct correspondence with one or more ontology design patterns. These patterns are also described in Montiel-Ponsoda [6]. They help users in the development of ontologies by using a system that permits an automatic detection of the ontological relations expressed as requirements by the users or domain experts. However, the analysis carried out by the authors is not exhaustive, and even though they support sentences, they

do not support requirements written as competency questions.

Regarding the translation of requirements into a formal language, the work presented by Aguado de Cea et al. [7] proposes a semi-automatic reuse of ontology design patterns that takes as input formulations in natural language of the domain aspect to be modelled, and obtains as output a set of ontology design patterns for solving the initial ontological needs. More focused on fully automating the formalization process, the work presented by Zemmouchi-Ghomari and Ghomari [5] and the work presented by Bezerra and Freitas [1] propose two different approaches to translate requirements into SPARQL queries using natural language processing. However, these works are limited to requirements which follow patterns such as "*What is...?*" or "*How much..?*", which represent a small subset of the patterns that are used in the definition of requirements. A more recent work about translating competency questions into SPARQL queries and that provides a broad analysis in requirements' patterns is the one presented by Ren and colleagues [4]; in this work, the authors provide a set of competency question patterns and a set of types of tests extracted from them that can be translated into SPARQL queries. Finally, the work proposed by Wiśniewski and colleagues [13] extends the competency question patterns proposed by Ren and colleagues and translates them into SPARQL-OWL queries.

After the review of these works, we conclude that there are no works devoted to analyse the difference between what is defined in the requirements and what is actually defined in the ontology in order to ascertain the degree of completeness of the requirements specification activity.

VII. Conclusions

This work conducts an empirical analysis about how requirements are defined in real-world standard and non-standard ontologies, analysing to what extent the axioms in the ontology are deduced from the requirements or if they are actually product of the ontology engineers' knowledge.

From the experimental study, we found that the analysed requirements were vague, and they only cover a small set of OWL constructs. In fact, it was determined that the ontology engineers take a considerable amount of modelling decisions that are not deduced from the requirements in order to complete the model. If the requirements were more accurate, they could reduce these modelling decisions and, therefore, reduce the risk of including errors in the modelling.

It must be taken into account that, during the process of collecting requirements, significant difficulties to find available real-world functional requirements were found. It should be pointed out that, especially for standard ontologies, the ontological requirements should be openly available in order to be able to carry out tests or to verify conformance between ontologies. These requirements can be helpful not only for ontology engineers to develop the ontologies, but also to users in order to be aware of the requirements covered by such ontologies.

Future work will be directed to a further analysis on the translation between requirements to particular ontology elements for the selected ontologies based on the extended corpus of categorised requirements introduced in this paper. This analysis would identify whether there are patterns that can help the automatic generation of draft ontology implementations from these requirements. In addition, a deeper analysis on how the ontology elicitation activity should be accomplished will also be considered.

Acknowledgements: This work is partially supported by the H2020 projects VICINITY: Open virtual neighbourhood network to connect intelligent buildings and smart objects (H2020-688467) and OntoCommons: Ontology-driven data documentation for Industry Commons (H2020-958371), by ETSI Specialist Task Force 534, by KnowledgeSpaces: Técnicas y herramientas para la gestión de grafos de conocimientos para dar soporte a espacios de datos (PID2020-118274RB-I00) and by a Predoctoral grant from the I+D+i program of the Universidad Politécnica de Madrid.

References

[1] C. Bezerra, F. Freitas, and F. Santana, "Evaluating ontologies with competency questions," in *Web Intelligence (WI) and Intelligent Agent Technologies (IAT), 2013 IEEE/WIC/ACM International Joint Conferences on*, vol. 3. IEEE, 2013, pp. 284–285.

[2] E. Montiel-Ponsoda and G. Aguado de Cea, "Using natural language patterns for the development of ontologies," *Researching specialized languages*, pp. 211–230, 2010.

[3] M. Dennis, K. van Deemter, D. Dell'Aglio, and J. Z. Pan, "Computing Authoring Tests from Competency Questions: Experimental Validation," in *Proceedings of the 16th International Semantic Web Conference, Vienna, Austria, October 21–25, 2017*. Springer, 2017, pp. 243–259.

[4] Y. Ren, A. Parvizi, C. Mellish, J. Z. Pan, K. Van Deemter, and R. Stevens, "Towards competency question-driven ontology authoring," in *Proceedings of the 11th European Semantic Web Conference, Crete, Greece, May 25-29, 2014*. Springer, 2014, pp. 752–767.

[5] L. Zemmouchi-Ghomari and A. R. Ghomari, "Translating natural language competency questions into SPARQLQueries: a case study," in *The First International Conference on Building and Exploring Web Based Environments*, 2013, pp. 81–86.

[6] E. Montiel-Ponsoda, *Multilingualism in Ontologies - Building Patterns and Representation Models*. LAP Lambert Academic Publishing, 2011.

[7] G. A. De Cea, A. Gómez-Pérez, E. Montiel-Ponsoda, and M. C. Suárez-Figueroa, "Natural language-based approach for helping in the reuse of ontology design patterns," in *Proceedings of the 16th International Conference on Knowledge Engineering and Knowledge Management, Acitrezza, Italy, September 29 - October 2, 2008*. Springer, 2008, pp. 32–47.

[8] E. Daga, E. Blomqvist, A. Gangemi, E. Montiel-Ponsoda, N. Nikitina, V. Presutti, and V.-T. Boris, "NeOn D2. 5.2 Pattern based ontology design: methodology and software support."

[9] A. Fernández-Izquierdo, M. Poveda-Villalón, and R. García-Castro, "CORAL: A Corpus of Ontological Requirements Annotated with Lexico-Syntactic Patterns," in *Proceedings of 16th Extended Semantic Web Conference, Portorož, Slovenia, June 2-6, 2019*. Springer, 2019, pp. 443–458.

[10] A. Gangemi and V. Presutti, "Ontology design patterns," in *Handbook on ontologies*. Springer, 2009, pp. 221–243.

[11] M. C. Suárez-Figueroa and A. Gómez-Pérez, "Ontology requirements specification," in *Ontology Engineering in a Networked World*. Springer, 2012, pp. 93–106.

[12] M. Uschold and M. Gruninger, "Ontologies: Principles, methods and applications," *The knowledge engineering review*, vol. 11, no. 2, pp. 93–136, 1996.

[13] D. Wiśniewski, J. Potoniec, A. Ławrynowicz, and C. M. Keet, "Analysis of Ontology Competency Questions and their formalizations in SPARQL-OWL," *Journal of Web Semantics*, vol. 59, p. 100534, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1570826819300617

Modelos gráficos probabilísticos basados en mixturas para el problema de partial label ranking

Juan C. Alfaro
Departamento de Sistemas Informáticos
Universidad de Castilla-La Mancha
Albacete, Spain
JuanCarlos.Alfaro@uclm.es

Juan A. Aledo
Departamento de Matemáticas
Universidad de Castilla-La Mancha
Albacete, Spain
JuanAngel.Aledo@uclm.es

José A. Gámez
Departamento de Sistemas Informáticos
Universidad de Castilla-La Mancha
Albacete, Spain
Jose.Gamez@uclm.es

***Resumen*—El problema de *Label Ranking* consiste en aprender *modelos de preferencia* a partir de conjuntos de datos etiquetados con un *ranking* de las etiquetas de la variable clase, con el objetivo de predecir un ranking para nuevas instancias. En este trabajo, abordamos el caso particular en el que tanto los rankings del conjunto de entrenamiento como los predichos pueden contener *empates*, lo que se conoce como el problema de *Partial Label Ranking*. En particular, proponemos *modelos gráficos probabilísticos* para resolver este problema. Dada la ausencia de una distribución de probabilidad para modelar rankings con empates, transformamos el espacio de rankings a un conjunto de variables discretas que expresan directamente las relaciones de precedencia (*precede*, *empata*, *sucede*) entre pares de etiquetas de la variable clase (distribución *multinomial*). En esta propuesta proponemos usar una *red bayesiana* con estructura Naive Bayes y una *variable oculta* como *raíz* de la misma, que será la encargada de recoger las interacciones entre las distintas variables (predictoras y objetivo). La inferencia se diseña en dos pasos: primero se obtienen las probabilidades a posteriori usando la red y estas se usan como entradas de la matriz de preferencias sobre la que se resuelve el problema de agregación de rankings. La evaluación experimental muestra que las propuestas realizadas son competitivas en términos de tasa de acierto con los algoritmos estándar basados en los *vecinos más cercanos* (*Instance Based Partial Label Ranking*) y *árboles de decisión* (*Partial Label Ranking Trees*).**

***Palabras clave*—Modelos basados en mixturas; Redes bayesianas; Naive Bayes; (Partial) Label Ranking**

I. Introducción

En los últimos años ha crecido el interés por los problemas de *clasificación supervisada no estándar*, entre los que se encuentra el problema de *Label Ranking* (*LR*) [9]. Este consiste en aprender *modelos de preferencia* que predicen *rankings* (t.c.c. *órdenes totales* o *permutaciones*) definidos sobre las etiquetas de la variable clase. Una diferencia destacable entre este problema y otros de clasificación no estándar (p.e., *clasificación ordinal*) es que las instancias del conjunto de datos de entrenamiento vienen etiquetadas con rankings, no con una única etiqueta, y estos se utilizan para el aprendizaje del modelo.

En este trabajo, nos centramos en el problema de *Partial Label Ranking* (*PLR*) [6], en el que tanto los rankings asociados a las instancias del conjunto de datos de entrenamiento como las predicciones dadas como salida son *rankings parciales* (t.c.c. *órdenes completos con empates* o *bucket orders*), esto es, un ranking con (posibles) *empates* entre las etiquetas de la variable clase.

El objetivo de este trabajo es resolver el problema de PLR mediante *modelos gráficos probabilísticos* (*PGMs*) [20]. Basándonos en [23], usamos una *red bayesiana híbrida* [15] con una *variable discreta oculta* como nodo raíz para modelar conjuntamente las distribuciones de probabilidad de los atributos (*multinomial* para discretos y *gaussiana* para continuos) y rankings. Es importante destacar que si bien es posible modelar probabilísticamente las permutaciones (rankings completos sin empates) mediante la *distribución de Mallows* [21], no conocemos distribución alternativa para modelar un conjunto de órdenes parciales. En su lugar, nuestra propuesta consiste en transformar la preferencia global representada por un ranking en un conjunto de preferencias parciales entre pares de etiquetas (*precede*, *empata*, *sucede*). La ventaja es que estas variables son de naturaleza discreta y, por tanto, fácilmente integrables en la red bayesiana mediante la distribución multinomial. La predicción obtenida para estas variables objetivo será la base para resolver el *problema de agregación de rankings* correspondiente [13] y poder así predecir un ranking parcial para la instancia de entrada.

El resto del artículo se organiza como sigue. En la Sección II, repasamos algunos conceptos básicos para tratar con rankings. En la Sección III, describimos formalmente el modelo propuesto. En la Sección IV, extendemos este modelo para permitir interacciones entre los atributos (continuos) mediante una distribución gaussiana multivariada. En la Sección V, presentamos la experimentación llevada a cabo para evaluar los métodos propuestos. Por último, en la Sección VI, proporcionamos las conclusiones y posibles trabajos futuros.

II. Agregación de rankings

Dado un conjunto de *ítems* $[[n]] = \{1, \ldots, n\}$, un *ranking* π representa una relación de precedencia entre estos. En particular, los rankings pueden ser *sin empates* (*t.c.c. órdenes totales* o *permutaciones*) o *con empates* (t.c.c. *rankings parciales*, *órdenes completos con empates* o *bucket orders*) si no hay preferencia entre algunos de los ítems rankeados.

El *problema de agregación de rankings* (*RAP*) [1] consiste en obtener un *orden de consenso* a partir de un conjunto de

rankings. En particular, el *Kemeny Ranking Problem* (*KRP*) [19] es probablemente el RAP más estudiado, y consiste en obtener la *permutación de consenso* (t.c.c. *permutación central*) que minimice la distancia (típicamente, la *distancia de Kendall*) con respecto a un conjunto de permutaciones. El algoritmo de *conteo de Borda* [11] se usa a menudo para resolver el KRP por su sencillez y rapidez.

Otro RAP que ha sido ampliamente estudiado en la literatura es el *Optimal Bucket Order Problem* (*OBOP*) [13]. El objetivo del OBOP es obtener la *matriz de buckets* B asociada a un bucket order π que minimize la distancia D

$$D(B,P) = \sum_{u,v \in [[n]]} |B(u,v) - P(u,v)|$$

donde P es la *matriz de ordenación por pares* asociada a un conjunto de bucket orders (ver [13] para los detalles).

Si bien existen varios algoritmos heurísticos para resolver el OBOP p.e. [2]–[4], [18], en este trabajo usaremos una instancia particular de *Bucket Pivot Algorithm with least indecision assumption* [2] denominada LIA_G^{MP2}, dado su buen balance entre precisión y tiempo.

III. Hidden Naive Bayes

Puesto que el problema a resolver es obtener un ranking parcial o bucket order para una instancia de entrada, necesitamos obtener la matriz de ordenación por pares P sobre la que se resolverá el RAP correspondiente. Nuestra propuesta es usar una red bayesiana para obtener los valores de cada una de las entradas de esa matriz, las cuales codifican la preferencia de la etiqueta c_u sobre la etiqueta c_v, con $u < v$. Puesto que $P(u,v) = 1 - P(v,u)$, podemos modelar ambas celdas mediante una única variable $Z_{u,v}$, en la que además por la naturaleza del ranking a predecir modelaremos de forma explícita la posibilidad del empate. Así, para cada par de etiquetas de la variable clase tendremos la variable discreta:

$$Z_{u,v} = \begin{cases} z_1 & \text{si} \quad c_u \succ c_v \\ z_2 & \text{si} \quad c_u \sim c_v \\ z_3 & \text{si} \quad c_u \prec c_v \end{cases}$$

Este tratamiento de las preferencias tiene como ventaja la simplicidad de manejar variables discretas, pero la complejidad de que el número de variables objetivo crece cuadráticamente con el número de etiquetas. Específicamente, tendremos que modelar $n_L = (n \cdot (n-1))/2$ variables objetivo.

Para obtener la probabilidad *a-posteriori*, dada una instancia, de las n_L variables objetivo, proponemos usar una red bayesiana con estructura Naive Bayes y con una variable oculta para capturar las interacciones entre las variables predictoras y objetivo.

A. Definición del modelo

La Figura 1 muestra la estructura propuesta usando la notación de *Plateau*. Como podemos ver, en la red (híbrida) conviven variables predictoras discretas y continuas con las variables objetivo de naturaleza discreta y la variable oculta H, también de naturaleza discreta. En particular, tendremos:

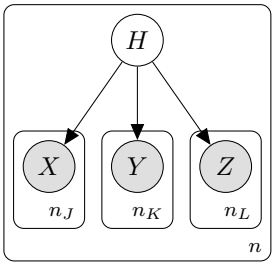


Fig. 1. Modelo HNB propuesto.

- *Variables predictoras discretas*, denotadas por X_j, $j = 1, \ldots, n_J$ con $dom(X_j) = \{x_{j_1}, \ldots, x_{j_{r_j}}\}$. Observadas en tiempo de aprendizaje e inferencia.
- *Variables predictoras continuas*, denotadas por Y_k, $k = 1, \ldots, n_K$. Observadas en tiempo de aprendizaje e inferencia.
- *Variables objetivo*, denotadas por $Z_{u,v}$, $u = 1, \ldots, n-1$ y $v = u+1, \ldots, n$, cuyo dominio es $dom(Z_{u,v}) = \{z_1, z_2, z_3\}$, siendo $z_1 = c_u \succ c_v$, $z_2 = c_u \sim c_v$ y $z_3 = c_u \prec c_v$. Observadas únicamente en tiempo de aprendizaje.
- *Variable oculta*, denotada por H con $dom(H) = \{h_1, \ldots, h_{r_H}\}$, donde r_H es el número total de componentes que tendrá la mixtura. El valor de esta variable nunca es conocido.

La variable oculta sigue una distribución multinomial, los atributos discretos y las variables objetivo (relaciones de precedencia en el ranking) siguen una distribución multinomial (condicionada a H) y los atributos continuos siguen una distribución gaussiana (condicionada a H). La distribución de probabilidad conjunta queda por tanto factorizada como sigue:

$$P(H, X_1, \ldots, X_{n_J}, Y_1, \ldots, Y_{n_K}, Z_{1,2}, \ldots, Z_{n-1,n}) =$$
$$p(h_w) \cdot \prod_{j=1}^{n_J} P(X_j|H) \cdot \prod_{k=1}^{n_K} P(Y_k|H) \cdot \prod_{u=1, v=u+1}^{u=n-1, v=n} P(Z_{u,v}|H)$$

B. Estimación de los parámetros

Dado que tanto las variables predictoras como la variable ranking no contienen valores perdidos y son *independientes e idénticamente distribuidas*, solo tenemos que tratar con la variable oculta H. Por tanto, usamos el algoritmo *Expectation-Maximization* (*EM*) para estimar conjuntamente los parámetros de las variables observadas y latentes:

- **Paso E**: Bajo la suposición de que los parámetros de los atributos discretos $p(x_{j_i}^{h_w})$, atributos continuos $\mu_k^{h_w}, \sigma_k^{h_w}$, variable ranking $p(z_{u,v}^{h_w})$ y variable oculta $p(h_w)$, $j = 1, \ldots, n_J$, $i = 1, \ldots, r_j$, $k = 1, \ldots, n_K$, $l = 1, \ldots, n_L$, $u = 1, \ldots, n-1$, $v = u+1, \ldots, n$, $w = 1, \ldots, r_H$ son conocidos, la probabilidad de que un ejemplo $e_t = (x_{1,t}, \ldots, x_{n_J,t}, y_{1,t}, \ldots, y_{n_K,t}, \pi_t)$ esté en una mixtura es

$$P(h_w|x_{1,t},\ldots,x_{n_J,t},y_{1,t},\ldots,y_{n_K,t},\pi_t) = \frac{1}{C}\cdot p(h_w)\cdot\prod_{k=1}^{n_K}\frac{1}{\sigma_{k,t}^{h_w}\sqrt{2\pi}}\cdot exp\left(-\frac{1}{2}\left(\frac{y_{k,t}-\mu_{k,t}^{h_w}}{\sigma}\right)^2\right)\cdot \prod_{j=1}^{n_J}p(x_{j,t}^{h_w})\cdot\prod_{u=1,v=u+1}^{u=n-1,v=n}p(z_{u,v,t}^{h_w}) \quad (1)$$

donde C es la constante de normalización.

- **Paso M**: Bajo la suposición de que la probabilidad de pertenecer a cada mixtura para todas las muestras es conocida, los parámetros se pueden estimar como sigue:
  - Parámetros multinomiales para los atributos discretos y variables objetivo (ranking). Cada parámetro multinomial se estima por *máxima verosimilitud* (*MLE*), pesando la frecuencia para cada instancia de acuerdo con la probabilidad de $H = h_w$ dada la instancia.
  - Parámetros gaussianos para los atributos continuos. Los parámetros gaussianos $\mu_k^{h_w}$ y $\sigma_k^{h_w}$ se estiman mediante MLE para cada $H = h_w$, pesando cada instancia por la probabilidad de pertenecer a la mixtura.

Criterio de parada: Usamos la *log-verosimilitud* del modelo dados los datos con un valor de convergencia $\alpha = 0.001$. Además, para evitar que el tiempo de entrenamiento del modelo crezca de manera excesiva, hemos impuesto un máximo de $\beta = 100$ iteraciones.

C. Aprendizaje: Selección de modelos

Además de la estructura gráfica y la estimación de parámetros, necesitamos un criterio para determinar la estructura interna de H, esto es, el número de estados. Para ello, seguimos el siguiente procedimiento:

1) Comenzamos dividiendo el conjunto de datos en entrenamiento Tr y validación Tv con una proporción de 0.8 y 0.2, respectivamente, y usamos el *coeficiente de correlación* τ_X para evaluar los modelos durante la búsqueda.
2) El número de mixturas se busca mediante un procedimiento iterativo en dos fases. Durante la primera fase, se evalúa el modelo con $r_H = 2^1, 2^2, 2^4, \ldots, 2^{10}$ y se selecciona el mejor valor r'_H de acuerdo con τ_X^{Tv}. En la segunda fase, se lleva a cabo una búsqueda binaria en el rango $[\frac{r'_H}{2}, r'_H]$. Finalmente, se utiliza el mejor valor r_H^* encontrado durante la búsqueda para entrenar el modelo con todo el conjunto de datos.

Cada vez que se prueba un nuevo valor para r_H se inicia el procedimiento desde cero, esto es, se inicializan los parámetros de las componentes (probabilidades y pesos) usando el *método de agrupamiento k-medias* [24] con $k = r_H$ y $\gamma = 10$ reinicializaciones. Tras ello, se ejecuta el algoritmo EM.

D. Inferencia

Durante la fase de inferencia se tiene que determinar el ranking (bucket order) π_t de los valores en $dom(C)$ asociado a una instancia de entrada e_t. Si bien el enfoque estándar es seleccionar el ranking que maximiza la *probabilidad a-posteriori* (obtenida marginalizando las variables) dada una muestra e_t, la alta cardinalidad del espacio de búsqueda, aproximadamente

$$\frac{n!}{2\cdot\ln 2^{n+1}},$$

fuerza a usar un método aproximado:

1) Calculamos la probabilidad a posteriori $P(Z_{u,v}|e_t)$ para cada variable objetivo mediante inferencia en la red bayesiana. Debido a su estructura esto puede hacerse en tiempo lineal.
2) Rellenamos las entradas de la matriz de ordenación por pares P_t para la instancia e_t como:

$$P_t(u,v) = P(Z_{u,v} = z_1|e_t) + \tfrac{1}{2}\cdot P(Z_{u,v} = z_2|e_t)$$

$$P_t(v,u) = 1 - P_t(u,v) = P(Z_{u,v} = z_3|e_t) + \tfrac{1}{2}\cdot P(Z_{u,v} = z_2|e_t)$$

con $u < v$ y $P_t(u,v) = 0.5$ si $u = v$, y resolvemos el OBOP sobre P_t para obtener el ranking π_t predicho para la instancia e_t.

IV. Gaussian Mixture Semi Naive Bayes

En esta sección, asumimos que todas las variables predictoras son continuas (es el caso de los *benchmarks* en PLR) y extendemos la complejidad del modelo permitiendo interacciones entre las variables predictoras, no así entre las variables objetivo.

A. Definición

Proponemos un modelo tipo *Semi-Naive Bayes* (*SNB*) [8], [16] en el que las variables continuas serán modeladas explícitamente mediante una distribución normal multivariada en lugar del producto de gaussianas independientes. El resto de interacciones sigue manejándose de forma implícita mediante la variable oculta H.

Consideraremos dos variantes del *modelo gaussiano basado en mixturas* (*GMM*) [22]: *completa*, donde cada mixtura estima una matriz de covarianzas, y *empatada*, que asume una misma matriz de covarianzas para todas las mixturas.

B. Estimación

Las principales diferencias respecto al modelo HNB son:

- **Paso E**: Es idéntico al descrito para HNB, pero sustituyendo, en la Ecuación 1, el producto de las gaussianas condicionales para las variables predictoras continuas por la función de densidad de probabilidad de la distribución normal multivariada $\mathcal{MN}(\vec{y}_t|\vec{\mu}^{h_w}, \Sigma^{h_w})$, siendo $\vec{y}_t$ la configuración de valores para las variables continuas en e_t, $\vec{\mu}^{h_w}$ el vector de medias y Σ^{h_w} la matriz de covarianzas.
- **Paso M**: Al igual que para el resto de parámetros (en HNB), las medias y covarianzas empíricas de los atributos continuos del modelo se calculan de manera estándar, esto es, pesando cada instancia por $w_t^{h_w} = P(h_w|\vec{y}_t, \pi_t)$.

TABLA I
DESCRIPCIÓN DE LOS CONJUNTOS DE DATOS.

Conjunto de datos	#instancias	#atributos	#etiquetas	#rankings	#buckets
authorship	841	70	4	47	3.063
blocks	5472	10	5	116	2.337
breast	109	9	6	62	3.925
ecoli	336	7	8	179	4.140
glass	214	9	6	105	4.089
iris	150	4	3	7	2.380
letter	20000	16	26	15014	7.033
libras	360	90	15	356	6.889
pendigits	10992	16	10	3327	3.397
satimage	6435	36	6	504	3.356
segment	2310	18	7	271	3.031
vehicle	846	18	4	47	3.117
vowel	528	10	11	504	5.739
wine	178	13	3	11	2.680
yeast	1484	8	10	1006	5.929

C. Aprendizaje e inferencia

El aprendizaje e inferencia para el modelo GMSNB son idénticos a los propuestos para HNB, con la salvedad de usar la distribución normal multivariada para el modelado de las variables predictoras continuas.

V. EVALUACIÓN EXPERIMENTAL

Detallamos aquí los conjuntos de datos utilizados, los algoritmos evaluados, la metodología adoptada y los resultados obtenidos en la evaluación de nuestra propuesta.

A. Conjuntos de datos

La Tabla I muestra las principales características de los 15 conjuntos de datos (semi-sintéticos) utilizados como *benchmark* en el problema de PLR [6]. Las columnas #rankings y #buckets son el número de bucket orders (rankings) diferentes en el dataset y el número medio de buckets por ranking. Los conjuntos de datos (y su descripción) se proporcionan en: https://www.openml.org/u/25829.

B. Algoritmos

- *Instance Based Partial Label Ranking* (*IBPLR*) [6]. Se ha utilizado la distancia Euclídea para identificar los k vecinos más cercanos, y pesado los bucket orders asociados a dichas instancias usando la inversa de la distancia. El número de vecinos más cercanos se ha ajustado usando una cinco validación cruzada (5-cv) sobre el conjunto de datos de entrenamiento (ver [6] para los detalles).
- *Partial Label Ranking Trees* (*PLRT*) usando los cuatro criterios de partición introducidos en [6].
- *HNB-PLR* (Sección III). Se han considerado cuatro instancias de este algoritmo: usando la distribución gaussiana (HNB-PLR-G) para las variables continuas y usando la distribución multinomial una vez se han discretizado las variables continuas en *igual anchura* (HNB-PLR-W), *igual frecuencia* (HNB-PLR-F) y *basado en entropía* [14] (HNB-PLR-E). En igual anchura e igual frecuencia se han considerado 5 intervalos.
- *GMSNB-PLR* (Sección IV), usando una matriz de covarianzas diferente para cada mixtura (completa, GMSNB-PLR-F) y una misma matriz de covarianzas para todas las componentes (empatada, GMSNB-PLR-T).

TABLA II
TASA DE ACIERTO MEDIA PARA LOS ALGORITMOS HNB-PLR Y GMSNB-PLR.

Conjunto de datos	HNB-PLR-G	HNB-PLR-F	HNB-PLR-W	HNB-PLR-E	GMSNB-PLR-F	GMSNB-PLR-T
authorship	$\mathbf{0.814 \pm 0.020}$	0.797 ± 0.023	0.793 ± 0.023	0.797 ± 0.027	0.724 ± 0.022	0.806 ± 0.023
blocks	0.931 ± 0.005	0.926 ± 0.005	0.899 ± 0.008	$\mathbf{0.942 \pm 0.005}$	0.922 ± 0.007	0.926 ± 0.006
breast	0.736 ± 0.057	$\mathbf{0.760 \pm 0.055}$	0.648 ± 0.088	0.729 ± 0.058	0.641 ± 0.109	0.717 ± 0.076
ecoli	$\mathbf{0.758 \pm 0.035}$	0.728 ± 0.033	0.727 ± 0.032	0.740 ± 0.034	0.714 ± 0.035	0.757 ± 0.028
glass	0.692 ± 0.061	0.757 ± 0.045	0.707 ± 0.055	0.759 ± 0.049	0.662 ± 0.062	$\mathbf{0.761 \pm 0.043}$
iris	0.874 ± 0.058	0.860 ± 0.076	0.887 ± 0.043	0.802 ± 0.105	0.871 ± 0.046	$\mathbf{0.897 \pm 0.041}$
letter						
libras	0.579 ± 0.031	0.545 ± 0.027	0.558 ± 0.034	$\mathbf{0.613 \pm 0.034}$	0.289 ± 0.030	0.578 ± 0.039
pendigits	0.807 ± 0.005	0.804 ± 0.006	0.806 ± 0.005	0.805 ± 0.005	0.793 ± 0.007	$\mathbf{0.809 \pm 0.006}$
satimage	0.870 ± 0.006	0.857 ± 0.007	0.843 ± 0.007	0.857 ± 0.007	0.813 ± 0.009	$\mathbf{0.875 \pm 0.006}$
segment	0.866 ± 0.013	0.867 ± 0.009	0.870 ± 0.011	$\mathbf{0.883 \pm 0.009}$	0.846 ± 0.013	0.871 ± 0.012
vehicle	0.731 ± 0.030	0.727 ± 0.028	0.709 ± 0.029	0.697 ± 0.028	0.606 ± 0.043	$\mathbf{0.781 \pm 0.021}$
vowel	0.707 ± 0.032	0.728 ± 0.021	0.725 ± 0.024	0.562 ± 0.063	0.596 ± 0.027	$\mathbf{0.756 \pm 0.014}$
wine	0.835 ± 0.042	0.822 ± 0.051	0.821 ± 0.055	0.826 ± 0.047	0.824 ± 0.055	$\mathbf{0.850 \pm 0.047}$
yeast	0.747 ± 0.017	0.740 ± 0.014	0.715 ± 0.014	0.707 ± 0.027	0.731 ± 0.017	$\mathbf{0.775 \pm 0.011}$

C. Metodología

Se han tomado las siguientes decisiones:

- Se ha considerado la validación estándar en el problema de PLR, esto es, una 5×10 validación cruzada.
- La tasa de acierto se ha medido usando el coeficiente de correlación entre rankings τ_X [12].
- Para analizar adecuadamente los resultados, hemos llevado a cabo el procedimiento estadístico estándar [10], [17] mediante la herramienta `exreport` [7]. Este procedimiento se divide en dos:
  1) Primero, se lleva a cabo un *test de Friedman* con un nivel de significancia $\alpha = 0.05$. Si el p-valor obtenido $\leq \alpha$, se rechaza la hipótesis nula H_0 y se concluye que al menos un algoritmo no es igual al resto.
  2) Segundo y asumiendo que se ha rechazado H_0 en el test de Friedman, se aplica un post-hoc test mediante el *procedimiento de Holm* para descubrir los algoritmos que destacan sobre el resto. Este método compara todos los algoritmos contra el de control, esto es, el rankeado primero por el test de Friedman.

D. Reproducibilidad

Se proporciona el código fuente: https://github.com/alfaro96/scikit-lr. Los experimentos se han ejecutado en ordenadores con el sistema operativo CentOS Linux 7, una CPU Intel(R) Xeon(R) E5–2630 a 2.40GHz, y con 16GB de memoria RAM.

E. Resultados

En esta sección, analizamos los resultados obtenidos. Nos centramos en la tasa de acierto y el tiempo de CPU.

1) Tasa de acierto: Comenzamos por analizar los resultados obtenidos por los PGMs basados en mixturas. La Tabla II muestra la tasa de acierto (coeficiente de correlación de rankings τ_X) para esta familia de algoritmos. Cada celda contiene la media y desviación estándar sobre los conjuntos de datos de test en la 5×10-cv para el coeficiente de correlación de rankings τ_X entre los bucket orders reales y predichos (en negrita los mejores resultados para cada conjunto de datos).

TABLA III
RESULTADOS DEL TEST POST-HOC PARA LA TASA DE ACIERTO MEDIA DE LOS ALGORITMOS HNB-PLR Y GMSNB-PLR.

Método	ranking	*p*-valor	gana	empata	pierde
GMSNB-PLR-T	1.90	-	-	-	-
HNB-PLR-G	2.47	$\mathbf{4.068e^{-1}}$	9	0	6
HNB-PLR-E	3.27	$\mathbf{9.087e^{-2}}$	10	0	5
HNB-PLR-F	3.70	$2.525e^{-2}$	13	1	1
HNB-PLR-W	4.40	$1.010e^{-3}$	15	0	0
GMSNB-PLR-F	5.27	$4.148e^{-6}$	14	0	1

TABLA IV
TASA DE ACIERTO MEDIO PARA LOS ALGORITMOS IBPLR Y LAS VARIANTES DE PLRT.

Conjunto de datos	IBPLR	PLRT-A	PLRT-D	PLRT-E	PLRT-G
authorship	$\mathbf{0.829 \pm 0.018}$	0.757 ± 0.025	0.763 ± 0.019	0.780 ± 0.023	0.776 ± 0.023
blocks	0.937 ± 0.005	0.940 ± 0.004	0.941 ± 0.005	0.944 ± 0.004	$\mathbf{0.946 \pm 0.004}$
breast	0.751 ± 0.058	0.770 ± 0.075	$\mathbf{0.777 \pm 0.067}$	0.766 ± 0.058	0.763 ± 0.064
ecoli	0.759 ± 0.027	0.758 ± 0.027	$\mathbf{0.765 \pm 0.026}$	0.763 ± 0.033	0.755 ± 0.031
glass	0.756 ± 0.045	$\mathbf{0.764 \pm 0.048}$	0.761 ± 0.048	0.761 ± 0.037	0.758 ± 0.034
iris	0.900 ± 0.051	0.912 ± 0.046	0.909 ± 0.044	$\mathbf{0.916 \pm 0.045}$	0.905 ± 0.040
letter	$\mathbf{0.689 \pm 0.005}$	0.667 ± 0.004	0.667 ± 0.004	0.669 ± 0.005	0.670 ± 0.005
libras	$\mathbf{0.648 \pm 0.029}$	0.584 ± 0.029	0.588 ± 0.030	0.575 ± 0.026	0.583 ± 0.029
pendigits	$\mathbf{0.819 \pm 0.006}$	0.799 ± 0.006	0.801 ± 0.006	0.813 ± 0.006	0.811 ± 0.005
satimage	$\mathbf{0.881 \pm 0.005}$	0.834 ± 0.006	0.839 ± 0.006	0.846 ± 0.006	0.848 ± 0.005
segment	0.890 ± 0.008	0.886 ± 0.010	0.889 ± 0.009	0.894 ± 0.009	$\mathbf{0.896 \pm 0.009}$
vehicle	0.739 ± 0.020	0.747 ± 0.031	0.757 ± 0.027	$\mathbf{0.793 \pm 0.021}$	0.778 ± 0.021
vowel	$\mathbf{0.745 \pm 0.017}$	0.673 ± 0.031	0.680 ± 0.028	0.679 ± 0.023	0.682 ± 0.023
wine	$\mathbf{0.845 \pm 0.036}$	0.841 ± 0.050	0.837 ± 0.046	0.824 ± 0.046	0.825 ± 0.060
yeast	$\mathbf{0.790 \pm 0.010}$	0.769 ± 0.010	0.774 ± 0.009	0.775 ± 0.010	0.774 ± 0.009

Hemos basado nuestro análisis estadístico en el procedimiento descrito en la Sección V-C:

1) El *p*-valor obtenido en el test de Friedman ha sido 4.124e^{-6}, por lo que rechazamos la hipótesis nula (H_0) y, al menos, un algoritmo es diferente al resto.
2) La Tabla III muestra los resultados del test post-hoc tomando el algoritmo GMSNB-PLR-T como control. Las columnas *ranking* y *p*-valor representan el ranking obtenido por el test de Friedman y el *p*-valor ajustado por el procedimiento de Holm, respectivamente. Las columnas *gana*, *empata*, *pierde* hacen referencia con el número de veces que el algoritmo de control gana, empata y pierde con el algoritmo de la fila correspondiente. Los *p*-valores en negrita se corresponden con las hipótesis nulas (H_0) no rechazadas.

Del análisis podemos observar que el algoritmo GMSNB-PLR-T está rankeado primero y es estadísticamente diferente de HNB-PLR-F, HNB-PLR-W y GMSNB-PLR-F. No hay, sin embargo, diferencia estadística significativa con respecto a los modelos HNB-PLR-G y HNB-PLR-E. Respecto a GMSNB-LR-F, una posible explicación a su pobre rendimiento es el bajo número de instancias en los conjuntos de datos, que puede no ser suficiente para estimar correctamente las múltiples matrices de covarianzas cuando el número de componentes de la mixtura crece.

Para completar el análisis, vamos a comparar los mejores algoritmos (GMSNB-PLR-T, HNB-PLR-G y HNB-PLR-E) con respecto al método IBPLR y las distintas variantes de PLRT, cuyos resultados se muestran en la Tabla IV.

El análisis estadístico de estos resultados obtiene un *p-valor* 1.5e^{-2} para el test de Friedman, por lo que al menos un algoritmo es diferente al resto. Dado que el algoritmo IBPLR ha sido rankeado primero por el test de Friedman se toma como control para el test post-hoc de Holm, cuyo resultado se muestra en la Tabla V.

TABLA V
RESULTADOS DEL TEST POST-HOC PARA LA TASA DE ACIERTO MEDIA DE LOS ALGORITMOS.

Método	ranking	*p*-valor	gana	empata	pierde
IBPLR	3.07	-	-	-	-
PLRT-E	3.70	$\mathbf{5.763e^{-1}}$	8	0	7
PLRT-D	4.13	$\mathbf{5.763e^{-1}}$	9	0	6
PLRT-G	4.23	$\mathbf{5.763e^{-1}}$	9	0	6
PLRT-A	4.40	$\mathbf{5.442e^{-1}}$	9	0	6
GMSNB-PLR-T	4.63	$\mathbf{3.992e^{-1}}$	11	0	4
HNB-PLR-G	5.77	$1.523e^{-2}$	15	0	0
HNB-PLR-E	6.07	$5.574e^{-1}$	13	0	2

En base a estos resultados, podemos concluir que:

- El algoritmo IBPLR es el primero del ranking, si bien no hay diferencia estadística significativa con respecto a los algoritmos PLRT y a GMSNB-PLR-T. Sí hay diferencia respecto a los dos algoritmos que no consideran interacciones entre las variables predictoras (HNB-PLR-G y HNB-PLR-E). Estos resultados demuestran que los métodos basados en modelos son competitivos con respecto a los algoritmos basados en instancias para el problema de PLR, lo cuál no ocurría para el problema de LR [5], [9].
- Una de las principales desventajas de los algoritmos HNB-PLR y GMSNB-PLR es que tiene unos requisitos de memoria elevados, por lo que no son capaces de tratar con conjuntos de datos para los que se genera un gran número de variables objetivo, y tienen además un alto número de instancias. Es el caso de `letter` (con 325 variables objetivo y 20000 instancias), en el que no se han obtenido resultados (ver Tabla II).

2) Tiempo: Los algoritmos propuestos en este estudio son significativamente más lentos que los basados en instancias (IBPLR) y árboles (PLRT). Este hecho puede deberse al gran número de componentes de la mixtura (ver Tabla VI) necesarias para lograr un buen modelado de la distribución de probabilidad conjunta. Como consecuencia, el algoritmo EM debe iterarse un alto número de veces durante la selección de modelos. Además, al añadir un gran número de variables objetivo debido a la transformación realizada a partir del ranking, el algoritmo EM necesita mucho tiempo para converger. A modo de ejemplo, podemos decir que para el conjunto de datos `pendigits` (que tiene 10992 instancias y 10 etiquetas, esto es, 45 variables objetivo), el algoritmo GMSNB-PLR-T es 120 veces más lento que IBLR y más de 850 veces más lento que PLRT-G.

VI. CONCLUSIONES Y TRABAJO FUTURO

Se ha propuesto un algoritmo basado en redes bayesianas y agregación de rankings para abordar el problema de PLR. Puesto que no conocemos una distribución probabilística para

TABLA VI
NÚMERO MEDIO DE MIXTURAS DE CADA PGM.

Conjunto de datos	HNB-PLR-G	HNB-PLR-F	HNB-PLR-W	HNB-PLR-E	GMSNB-PLR-F	GMSNB-PLR-T
authorship	36.340 ± 36.988	24.58 ± 18.377	31.380 ± 21.813	40.840 ± 52.281	3.020 ± 0.141	35.420 ± 41.952
blocks	167.440 ± 76.169	238.400 ± 168.220	81.300 ± 33.121	341.660 ± 147.979	68.520 ± 23.693	213.200 ± 97.692
breast	15.960 ± 7.284	29.120 ± 19.157	19.800 ± 19.078	17.720 ± 12.795	4.520 ± 2.288	20.320 ± 8.110
ecoli	31.000 ± 14.321	25.820 ± 21.930	31.140 ± 26.869	39.900 ± 20.928	12.920 ± 6.552	47.040 ± 27.871
glass	17.220 ± 9.951	66.020 ± 32.922	27.920 ± 23.357	45.520 ± 23.603	5.180 ± 2.164	39.760 ± 16.577
iris	17.020 ± 15.946	34.820 ± 20.457	24.180 ± 17.253	9.560 ± 4.739	7.960 ± 4.000	32.320 ± 22.709
letter						
libras	47.660 ± 12.967	40.940 ± 14.621	43.960 ± 13.425	121.760 ± 38.154	218.080 ± 39.608	56.200 ± 10.900
pendigits	388.800 ± 108.298	204.680 ± 67.601	266.480 ± 95.088	261.520 ± 98.865	93.800 ± 27.355	405.840 ± 102.542
satimage	326.660 ± 101.758	283.660 ± 96.820	198.720 ± 108.040	272.060 ± 108.377	29.620 ± 14.246	392.460 ± 110.909
segment	140.400 ± 56.351	202.580 ± 158.267	196.300 ± 154.706	230.320 ± 141.072	42.680 ± 21.920	337.380 ± 121.548
vehicle	73.320 ± 35.361	292.600 ± 151.414	346.300 ± 144.204	172.420 ± 126.264	12.260 ± 2.448	66.480 ± 60.716
vowel	75.320 ± 26.250	169.260 ± 55.564	186.260 ± 49.278	95.640 ± 42.740	7.640 ± 2.884	174.580 ± 52.597
wine	6.700 ± 9.033	11.980 ± 15.213	17.120 ± 19.256	24.960 ± 23.206	3.800 ± 1.030	14.480 ± 17.117
yeast	103.500 ± 56.536	46.000 ± 18.553	127.660 ± 97.812	159.040 ± 113.482	30.300 ± 16.656	219.880 ± 93.535

modelar conjuntos de bucket orders, se ha planteado una transformación de la variable objetivo (ranking) en un conjunto de variables objetivo que modelan las preferencias entre pares de etiquetas. La ventaja es que al pasar a tratar con variables discretas, son modeladas con las distribución multinomial y fácilmente integrables en una red bayesiana. Nuestra propuesta se basa en un modelo tipo SNB con una variable oculta como raíz, que será la encargada de modelar las interacciones entre las variables. De esta forma la estructura del modelo es fija, pero es necesario recurrir al algoritmo EM para estimar los parámetros. Dada una instancia, la solución propuesta usa las probabilidades a-posteriori predichas para las variables objetivo como entrada para construir la matriz de preferencias que es usada por el algoritmo de agregación de rankings, obteniéndose así el ranking predicho para la instancia de entrada.

De la evaluación experimental realizada, concluimos que el algoritmo propuesto basado en modelar de forma conjunta las variables predictoras continuas mediante una distribución normal multivariada (GMSNB-PLR-T) es competitivo con los métodos disponibles en la literatura, esto es, IBPLR y PLRT. Es importante comentar que si bien el algoritmo GMSNB-PLR-T es más complejo computacionalmente que IBPLR durante la fase de aprendizaje, esto no ocurre durante la fase de inferencia, al ser IBPLR un método basado en instancias. Esto es importante en una gran parte de aplicaciones reales, pues normalmente requieren una fase de aprendizaje *off-line*, cuyo tiempo no suele ser tan relevante como el de inferencia, que se realiza *on-line*.

Observamos dos limitaciones importantes a nuestra propuesta. La primera es el número de variables objetivo (cuadrático en el número de etiquetas), lo que hace que los métodos tengan problemas de memoria cuando el número de etiquetas es grande. Como trabajo futuro pretendemos estudiar técnicas de agrupamiento que permitan descomponer el problema global en varios más pequeños. Creemos que esta línea de trabajo también redundará en una mejora respecto a la otra limitación observada, la gran cantidad de tiempo de CPU necesario para el entrenamiento (con respecto a los otros algoritmos evaluados).

VII. AGRADECIMIENTOS

Este trabajo ha sido parcialmente financiado por el Gobierno de España, fondos FEDER y la Junta de Comunidades de Castilla-La Mancha mediante los proyectos PID2019–106758GB–C33 /AEI/ 10.13039/501100011033 y SBPLY/17/180501/000493. Juan C. Alfaro también está financiado mediante la beca FPU18/00181 por el MCIU.

REFERENCIAS

[1] J. A. Aledo, J. A. Gámez, and D. Molina. Approaching the rank aggregation problem by local search-based metaheuristics. *Journal of Computational and Applied Mathematics*, 354:445–456, 2019.

[2] J. A. Aledo, J. A. Gámez, and A. Rosete. Utopia in the solution of the Bucket Order Problem. *Decision Support Systems*, 97:69–80, 2017.

[3] J. A. Aledo, J. A. Gámez, and A. Rosete. Approaching rank aggregation problems by using evolution strategies: The case of the optimal bucket order problem. *European Journal of Operational Research*, 270:982–998, 2018.

[4] J. A. Aledo, J. A. Gámez, and A. Rosete. A highly scalable algorithm for weak rankings aggregation. *Information Sciences*, 570:144–171, 2021.

[5] J.A. Aledo, J.A. Gámez, and D. Molina. Tackling the supervised label ranking problem by bagging weak learners. *Information Fusion*, 35:38–50, 2017.

[6] J. C. Alfaro, J. A. Aledo, and J. A. Gámez. Learning decision trees for the partial label ranking problem. *International Journal of Intelligent Systems*, 36:890–918, 2021.

[7] J. Arias and J. Cózar. ExReport: Fast, reliable and elegant reproducible research. http://jacintoarias.github.io/exreport (last accessed: June 18, 2021), 2016.

[8] C. Bielza and P. Larrañaga. Discrete Bayesian Network Classifiers: A Survey. *ACM Computing Surveys*, 47, 2014.

[9] W. Cheng, J. Hühn, and E. Hüllermeier. Decision Tree and Instance-Based Learning for Label Ranking. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 161–168, 2009.

[10] J. Demšar. Statistical Comparisons of Classifiers over Multiple Data Sets. *Journal of Machine Learning Research*, 7:1–30, 2006.

[11] P. Emerson. The original Borda count and partial voting. *Social Choice and Welfare*, 40:353–358, 2013.

[12] E. J. Emond and D. W. Mason. A new rank correlation coefficient with application to the consensus ranking problem. *Journal of Multi-Criteria Decision Analysis*, 11:17–28, 2002.

[13] R. Fagin, R. Kumar, M. Mahdian, D. Sivakumar, and E. Vee. Comparing and Aggregating Rankings with Ties. In *Proceedings of the Twenty-third ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, pages 47–58, 2004.

[14] U. M. Fayyad and K. B. Irani. Multi-interval Discretization of Continuous-Valued Attributes for Classification Learning. *Artificial Intelligence*, 13:1022–1027, 1993.

[15] A. Fernández, J. A. Gámez, R. Rumí, and A. Salmerón. Data clustering using hidden variables in hybrid Bayesian networks. *Progress in Artificial Intelligence*, 2:141–152, 2014.

[16] M.J. Flores, J. A. Gámez, and A. Martínez. Supervised Classification with Bayesian Networks: A Review on Models and Applications. In *Intelligent Data Analysis for Real-Life Applications: Theory and Practice*, pages 72–102. IGI Global, 2012.

[17] S. García and F. Herrera. An Extension on "Statistical Comparisons of Classifiers over Multiple Data Sets" for all Pairwise Comparisons. *Journal of Machine Learning Research*, 9:2677–2694, 2008.

[18] A. Gionis, H. Mannila, K. Puolamäki, and A. Ukkonen. Algorithms for Discovering Bucket Orders from Data. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 561–566, 2006.

[19] J. Kemeny and J. Snell. *Mathematical Models in the Social Sciences*. The MIT Press, 1972.

[20] D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques - Adaptive Computation and Machine Learning*. MIT Press, 2009.

[21] C. L. Mallows. Non-Null Ranking Models. *Biometrika*, 44:114–130, 1957.

[22] D. Reynolds. *Gaussian Mixture Models*, pages 659–663. Springer Science+Business, 2009.

[23] E. G. Rodrigo, J. C. Alfaro, J. A. Aledo, and J. A. Gámez. Mixture-Based Probabilistic Graphical Models for the Label Ranking Problem. *Entropy*, 23, 2021.

[24] X. Wu and V. Kumar. *The Top Ten Algorithms in Data Mining*. Chapman and Hall, 2009.

LabelDetection: Simplifying the Use and Construction of Deep Detection Models*

*Note: The full contents of this paper have been published in the volume Lecture Notes in Artificial Intelligence 12882 (LNAI 12882)

Ángela Casado-García, Jónathan Heras
Department of Mathematics and Computer Science
Universidad de La Rioja
La Rioja, Spain
{angela.casado, jonathan.heras}@unirioja.es

***Abstract*—Deep learning algorithms for object detection on images have been successfully applied in several fields; however, non-expert users might find difficult to adopt these techniques due to several reasons. First of all, using detection models requires some knowledge about the library employed to built them; and, in general, it is not usually possible to interact with the predictions produced by the models. In addition, training custom models is also challenging because there are several algorithms implemented in different libraries, and each of them uses its own annotation format, and is configured in a particular way. In this paper, we face all those challenges by developing LabelDetection, a graphical tool that allows non-expert users to employ models trained with different algorithms and libraries. Moreover, LabelDetection is an end-to-end application that provides the necessary features to annotate a dataset of images, and train a variety of object detection models. Finally, LabelDetection simplifies the use of advance techniques like test-time augmentation, that improves the accuracy of detection models; and data distillation, that reduces the number of images that must be annotated to train a model. LabelDetection is freely available at https://github.com/ancasag/LabelDetection.**

***Index Terms*—Deep learning, Object detection, Test time augmentation, Data distillation**

DOI: https://doi.org/10.1007/978-3-030-85713-4_2

Algoritmos para el problema del bandido multi-objetivo basados en la escalarización de Chebyshev

L. Mandow, S. Martín-Albo, J.L. Pérez de la Cruz
Universidad de Málaga, Andalucía Tech
Departamento de Lenguajes y Ciencias de la Computación
Málaga, España
lmandow@uma.es, perez@lcc.uma.es

***Resumen*—Este trabajo propone y analiza diversas adaptaciones de la escalarización de Chebyshev para la aproximación de la frontera de Pareto en el problema del bandido multi-objetivo (multi-objective bandit problem). Los algoritmos se evalúan sobre un problema de referencia con dos objetivos. Su rendimiento se analiza mediante tres medidas: la probabilidad de realizar una acción óptima, una medida de pérdida (regret), y una medida de inequidad (unfairness). Se contribuye un algoritmo que mejora en al menos un orden de magnitud la velocidad de convergencia sobre la anterior propuesta.**

***Index Terms*—Problema del bandido, decisión multi-objetivo, escalarización de Chebyshev**

I. Introducción

El problema del bandido [1] es una formalización matemática que tiene su origen en un problema de asignación de recursos para ensayos clínicos. Supongamos que tenemos varios tratamientos, y cada ensayo con uno de ellos tiene una probabilidad desconocida de terminar con éxito o fracaso. La cuestión es cómo determinar el mejor tratamiento con el menor número posible de ensayos.

Este problema puede asimilarse también a una máquina tragaperras (o tragamonedas) con distintas palancas. En inglés se denominan informalmente también estas máquinas como *bandidos* (bandits), y al problema como *bandido multi-brazo o multi-palanca* (multi-armed bandit). Cada palanca tiene asociada una distribución de probabilidad y devuelve una recompensa aleatoriamente de acuerdo con la misma. El objetivo es maximizar la recompensa esperada tras realizar un número dado de acciones. Este problema se considera un caso simplificado del problema general del aprendizaje por refuerzo [2], y su solución está relacionada con algoritmos como MCTS (Monte Carlo Tree Search) y sus variantes como UCT (UCB1 applied to Trees) [3].

Financiado por Plan Propio de Investigación de la Universidad de Málaga - Campus de Excelencia Internacional Andalucía Tech. L. Mandow está financiado por el proyecto Rhea P18-FR-1081 (MCI/AEI/FEDER, UE) Junta de Andalucía (cofinanciado con fondos FEDER), España. S. Martín-Albo está financiado con una Beca de Iniciación a la Investigación para estudiantes de grado y máster, I Plan Propio de Investigación y Transferencia de la Universidad de Málaga, España.

Recientemente se ha propuesto generalizar el problema del bandido al caso multi-objetivo [4], de manera que cada acción recibe un vector de recompensas en lugar de un sólo valor escalar. También se han analizado divesas variantes para resolver este problema, de manera que se puedan aprender de forma eficiente todas las políticas Pareto óptimas del problema. Una alternativa interesante es aplicar diversas escalarizaciones al problema multi-objetivo, para así poder aproximar la frontera de Pareto.

Drugan y Nowé [4] analizaron las escalarizaciones lineal y de Chebyshev aplicadas al problema del bandido multi-objetivo. Las escalarizaciones lineales sólo permiten encontrar un subconjunto de las soluciones de Pareto (las denominadas *soluciones soportadas*) por lo que no son las más adecuadas en muchos casos prácticos. La escalarización de Chebyshev permite en principio encontrar cualquier solución óptima de Pareto. Sin embargo, los resultados obtenidos mostraron un rendimiento muy pobre [4].

En este artículo revisamos la escalarización de Chebyshev aplicada al problema del bandido multi-objetivo, y proponemos algoritmos alternativos que pueden conseguir mejoras sustanciales en el rendimiento.

Este artículo se organiza de la siguiente manera. En la sección II repasamos los principales conceptos relacionados con el problema del bandido, y su generalización multi-objetivo, así como sobre la escalarización de Chebyshev. En la sección III describimos el algoritmo propuesto por [4], y proponemos varias mejoras. La sección IV describe un ejemplo y analiza el rendimiento de los algoritmos. Por último, se resumen las conclusiones y se plantean algunas posibles continuaciones de este trabajo.

II. Antecedentes

II-A. *El problema del bandido*

Formalmente, consideraremos un problema con k *acciones* numeradas de 1 hasta k, cada una asociada a una distribución de probabilidad con media $\mu_i, i = 1 \dots k$. Las distribuciones son estacionarias e independientes entre sí, y las distintas

ejecuciones de la misma acción devuelven muestras (recompensas) independientes e idénticamente distribuidas tomadas de su distribución. Las distribuciones son desconocidas para un agente que debe realizar N pasos secuencialmente eligiendo y realizando una acción en cada uno. Cada posible estrategia de decisión es una *política*.

Sea μ^* la media más alta, correspondiente a la acción óptima. Realizar una acción subóptima i tiene asociada una *pérdida esperada* (regret) $\Delta_i = \mu^* - \mu_i$. El objetivo del agente es *minimizar la pérdida esperada* al realizar los N pasos, o lo que es lo mismo, *maximizar la recompensa esperada*.

Para resolver el problema el agente debe ir construyendo estimaciones del valor esperado de cada acción, y resolver el conocido como *dilema entre explorar y sacar provecho* (exploration/exploitation dilemma). Eligiendo en cada momento la acción con mayor estimación, o *codiciosa* (greedy), el agente puede sacar provecho de la información acumulada para obtener una buena recompensa de forma inmediata. Sin embargo, explorar acciones menos prometedoras puede ayudar a obtener mejores estimaciones, que a su vez permitirán elegir acciones de manera más informada en el futuro.

Elegir siempre la acción codiciosa es generalmente una política subóptima. Lai y Robbins [5] demostraron que, de forma asintótica, lo mejor a lo que se puede aspirar es a una pérdida que crece de forma logarítmica con el número de acciones realizadas. También describieron políticas asintóticamente óptimas asociando un *índice de confianza superior* (upper confidence index) a cada acción, y eligiendo siempre la palanca con mayor índice.

Agrawal [6] mejoró este resultado proporcionando políticas asintóticamente óptimas más sencillas de calcular. Posteriormente, Auer, Cesa-Bianchi y Fischer [7] mejoraron los resultados anteriores describiendo políticas que consiguen una pérdida logarítmica de manera uniforme en el tiempo, utilizando también índices sencillos de calcular. La más conocida es la denominada política UCB1 (Upper Confidence Bound 1).

Los algoritmos 1 y 2 resumen el funcionamiento de UCB1 [7]. Básicamente, el algoritmo comienza realizando una vez cada acción, y calcula un promedio x_i de las recompensas obtenidas para cada una de ellas. En la línea 1 del algoritmo 2 se decide qué acción realizar. Esta corresponde a la que arroje una *cota superior* mayor, calculada como la suma de su promedio y un factor de confianza (sacar provecho). Este factor decrece con el número de veces n_k que se ha realizado cada acción, reflejando una mayor confianza en los promedios con mayor número de muestras, pero se incrementa también con el número total t de acciones realizadas. Por ello, cuando una acción lleva mucho tiempo sin ser elegida, el factor de confianza acabará haciendo de nuevo su cota atractiva, aunque sea brevemente (exploración). La línea 3 mantiene los promedios actualizados con cada nueva muestra.

Algorithm 1 Algoritmo UCB1. El método $ejecutar(a)$ realiza la acción a y devuelve la recompensa.

Entrada N, número de pasos; K, número de acciones; $\vec{a} = (a_1, \ldots, a_K)$, vector de acciones.

1: Definir $\vec{x} = (x_1, \ldots x_K), \vec{n} = (n_1, \ldots n_K)$ vectores para mantener la recompensa promedio y el contador de pasos para cada acción.
2: **for** $i \in \{1..K\}$ **do**
3: $\quad x_i \leftarrow ejecutar(a_i)$
4: $\quad n_i \leftarrow 1$
5: **end for**
6: $t \leftarrow K$
7: **for** N pasos **do**
8: $\quad pasoUCB1(\vec{a}, \vec{x}, \vec{n}, t)$
9: **end for**

Algorithm 2 Método pasoUCB1. El método $ejecutar(a)$ realiza la acción a y devuelve la recompensa.

Entrada $\vec{a}, \vec{x}, \vec{n}$, vectores de acciones, recompensas promedio y contadores de pasos por acción; t, contador de pasos global.

1: $i \leftarrow \arg\max_k (x_k + \sqrt{\frac{2 \ln t}{n_k}})$
2: $n_i \leftarrow n_i + 1$
3: $x_i \leftarrow x_i + \frac{1}{n_i}(ejecutar(a_i) - x_i)$
4: $t \leftarrow t + 1$

II-B. Optimización multiobjetivo

El análisis multi-objetivo [8] es una generalización del problema de optimización escalar, en el que se considera un conjunto de D objetivos independientes entre sí y que no se pueden medir de forma conjunta. Sin pérdida de generalidad, supondremos que se desean maximizar todos los objetivos. Técnicamente, a cada problema le corresponde un conjunto de soluciones factibles $\mathcal{X}$, y el conjunto de vectores de beneficio $Y \subset \mathbb{R}^D$ asociados a cada una de ellas. Por simplicidad, hablaremos aquí de los elementos de Y directamente como si fueran las soluciones al problema.

Al contrario que en el caso escalar, la solución a un problema multi-objetivo no viene dada generalmente por una solución única. Dados dos vectores $\vec{u}, \vec{v} \in \mathbb{R}^D$, decimos que $\vec{u}$ domina a $\vec{v}$ sii para todo componente i, $u_i \geq v_i$ y al menos para algún j, $u_j > v_j$. La relación de dominancia es únicamente parcial, es decir, dados dos vectores no siempre existe una relación de dominancia entre ellos. La solución a un problema multi-objetivo viene dada por los denominados *óptimos de Pareto*, es decir, aquellas soluciones asociadas a vectores no dominados por el de ninguna otra solución. El conjunto de soluciones óptimas define la denominada *frontera de Pareto*, que representa todos los posibles compromisos óptimos entre objetivos. Dado un conjunto de vectores Y, denominaremos $\mathcal{N}(Y)$ al subconjunto de vectores no dominados.

Una alternativa al cálculo de toda la frontera de Pareto es el cálculo de una única solución Pareto-óptima[1] que cumpla directamente con las preferencias subjetivas expresadas por el centro decisor. En algunos casos es posible realizar una *escalarización* de los vectores de beneficio, es decir, asociar

[1] O un subconjunto reducido de las mismas.

un único valor escalar a todas las soluciones. En tal caso es posible aplicar los algoritmos tradicionales de optimización de un sólo objetivo para resolver el problema.

La *escalarización de Chebyshev*[2] pertenece a una familia de estrategias basadas en elegir la solución cuyo vector *minimiza la distancia* a un punto de referencia dado. Una referencia frecuente suele ser el denominado *punto ideal* $\vec{\alpha} = (\alpha_1, \dots \alpha_D)$, donde α_i es el valor óptimo obtenido en la maximización escalar del objetivo i sobre el conjunto de soluciones posibles,

$$\alpha_j = \max_{\vec{y} \in Y} y_j \tag{1}$$

Más concretamente, la distancia a minimizar es la *distancia de Chebyshev* (o norma-l_∞), definida para un vector $\vec{y} = (y_1, \dots, y_D)$ como,

$$\max_k \{w_k | y_k - \alpha_k|\} \tag{2}$$

donde el vector de pesos $\vec{w} = (w_1, \dots w_D)$ denota las preferencias establecidas por el centro decisor.

Una propiedad fundamental de este enfoque es que para toda solución no dominada $\vec{y^*}$ existe un conjunto de pesos tal que $\vec{y^*}$ minimiza la distancia a $\vec{\alpha}$ [9]. Igualmente, dadas las soluciones a:

$$\begin{aligned} &\text{Min } \max_k \{w_k | y_k - \alpha_k|\} \\ &\text{Sujeto a } \vec{y} \in Y \end{aligned} \tag{3}$$

al menos una de ellas es no dominada (aunque también pueden existir otras soluciones igualmente óptimas al problema escalar que sean dominadas).

Es importante señalar que, aunque la escalarización de Chebyshev puede emplearse para encontrar una solución concreta al problema, también puede emplearse para calcular o aproximar toda la frontera de Pareto. Para ello puede resolverse repetidamente el problema escalarizado con vectores de pesos distintos.

II-C. Bandidos multiobjetivo

Una generalización natural del problema del bandido es suponer que cada acción proporciona un vector de recompensas distintas, correspondientes a un conjunto de objetivos a maximizar que son independientes entre sí y no se pueden medir conjuntamente. Este problema ha sido propuesto recientemente por [4] y denominado MOMAB (Multi-Objective Multi-Armed Bandit problem). En la medida de lo razonable, nuestro artículo utiliza una notación compatible con dicho trabajo.

En [4] se considera la aplicación de diversas estrategias de optimización multi-objetivo al problema MOMAB. Tal como se explicó en la sección II-B, la escalarización de Chebyshev es una alternativa que ofrece diversas ventajas. En primer lugar, es computacionalmente eficiente ya que no precisa de complejos cálculos de dominancia. En segundo lugar, permite encontrar una solución Pareto-óptima a partir de un conjunto de preferencias expresadas de forma intuitiva por el centro decisor. Además, generalmente puede resolverse adaptando algortimos de optimización escalar. Por último, puede utilizarse como algoritmo de base para aproximar o calcular toda la frontera de Pareto.

[2]En su transliteración inglesa, o Tchebycheff en la francesa.

II-D. Escalarización de Chebyshev para bandidos multiobjetivo

Tal como explicamos en la sección II-B, es posible encontrar un vector que maximiza la recompensa vectorial recibida minimizando la distancia de los vectores de beneficio a un punto de referencia dado.

En [4] se plantea adaptar el algoritmo UCB1 para resolver este problema. Sin embargo, UCB1 es un algoritmo diseñado para maximización, no para minimización. La adaptación pasa por reformular el problema definiendo un nuevo punto de referencia, de forma que sea posible encontrar los vectores que maximizan la recompensa recibida *maximizando* la distancia al nuevo punto de referencia.

Para reformular la escalarización de Chebyshev consideraremos como punto de referencia el denominado punto *nadir* $\vec{z} = (z_1, \dots, z_D)$ donde[3]:

$$z_j = \min_{\vec{y} \in \mathcal{N}(Y)} y_j \tag{4}$$

de hecho, en [4] proponen restar un pequeño valor positivo ϵ_j a cada componente. En este contexto, dado el nuevo punto de referencia $\vec{z}$, podemos definir la nueva función de escalarización ponderada:

$$f_T(\vec{y}) = \min_k \{w_k(y_k - z_k)\} \tag{5}$$

Entonces, dadas las soluciones al siguiente problema:

$$\begin{aligned} &\text{Max } f_T(\vec{y}) \\ &\text{Sujeto a } \vec{y} \in Y \end{aligned} \tag{6}$$

al menos una de ellas será un vector no dominado.

Con esta reformulación es posible adaptar el algoritmo UCB1 para resolver el problema 6. Abordamos esta cuestión en la siguiente sección.

III. Algoritmos

En [4] se propone un algoritmo para aproximar la frontera de Pareto de un MOMAB resolviendo un conjunto de problemas escalares que emplean la escalarización de Chebyshev descrita en la sección II-D. Las ideas centrales son las siguientes:

- Se dispone de un conjunto de S funciones de escalarización de Chebyshev $f^1, \dots f^S$, cada una asociada a un vector de pesos diferente.
- El algoritmo elige de manera aleatoria y uniforme la función de escalarización usada en cada paso.

[3]Nótese que este punto se define únicamente en términos de los vectores en $\mathcal{N}(Y)$.

- Las distintas funciones poseen estimaciones y contadores separados para las distintas acciones. En la práctica, esto equivale a realizar S optimizaciones UCB1 intercaladas pero independientes, una por cada función de escalarización.
- Una vez realizada la escalarización de las estimaciones de una acción, se le añade el factor de confianza correspondiente al número de veces que se seleccionó esa función y se aplicó dicha acción.

Estas ideas se han recogido en los algoritmos 3 y 4. Denominamos C0 al procedimiento resultante.

A continuaciópn proponemos diversas mejoras al algoritmo C0 recién descrito:

1. En primer lugar, no es necesario realizar S optimizaciones UCB1 completamente independientes (una para cada función de escalarización). Podemos promediar todas las muestras de cada acción i, independientemente de la función f^j elegida en cada paso. De este modo, todas las funciones aprovechan toda la experiencia previa en la realización de acciones.
2. En segundo lugar, recordemos que el algoritmo UCB1 suma el factor de confianza directamente al promedio estimado de cada acción para acotarlo superiormente. Por el contrario, el algoritmo C0 suma el factor de confianza a la *escalarización de los promedios* (algoritmo 4, línea 1). En este trabajo proponemos *aplicar la escalarización a las cotas de las estimaciones*, en lugar de acotar la escalarización de las estimaciones.
3. Por último, hacemos notar que en el proceso de selección de la acción, puede ocurrir que varias de ellas empaten en la maximización de las cotas. Tal como se comentó en la sección II-B, al menos una de ellas corresponderá a un punto no dominado. Por tanto, en caso de empate proponemos desempatar siempre por una acción cuya estimación sea no dominada.

A modo de resumen, consideraremos a continuación los siguientes algoritmos:

- C0: Intercala S optimizaciones UCB1 independientes, de modo que cada función mantiene sus propias estimaciones de los valores de las acciones. El factor de confianza se añade a la escalarización de las estimaciones.
- C1: Todas las funciones comparten las estimaciones de los valores de las acciones (propuesta 1). El factor de confianza se añade a la escalarización de las estimaciones. Véanse los algoritmos 5 y 6.
- C2: Todas las funciones comparten las estimaciones de los valores de las acciones (propuesta 1). El factor de confianza se añade a las estimaciones, y el resultado se escalariza (propuesta 2). Este algoritmo coincide con C1, sustituyendo la llamada a $pasoC1$ por una llamada a $pasoC2$ (algoritmo 7).

La propuesta 3 se incorpora en los tres algoritmos.

IV. Experimentos y análisis

En esta sección evaluamos los algoritmos descritos en la sección III al problema de referencia propuesto en [4], y comparamos su rendimiento.

El problema consta de 20 acciones diferentes que proporcionan recompensa sobre dos objetivos. Los promedios de las recompensas de cada acción son $\mu_1 = (0{,}55, 0{,}5), \mu_2 = (0{,}53, 0{,}51), \mu_3 = (0{,}52, 0{,}54), \mu_4 = (0{,}5, 0{,}57), \mu_5 = (0{,}51, 0{,}51), \mu_6 = (0{,}5, 0{,}5)$ y $\mu_k = (0{,}48, 0{,}48), k = 7 \ldots 20$. Los cuatro primeros son óptimos de Pareto. En todos los casos las recompensas se obtienen de distribuciones de Bernoulli. Para calcular la frontera de Pareto se utilizan once escalarizaciones definidas por los pesos $(1, 0), (0{,}9, 0{,}1), \ldots (0{,}1, 0{,}9), (0, 1)$. El valor ϵ_i restado al punto nadir fue 0.01. Los resultados mostrados se promedian sobre 250 agentes diferentes. Se realizó un millón de pasos con cada agente.

IV-A. *Medidas de rendimiento*

En nuestros experimentos evaluamos las tres medidas de rendimiento propuestas en [4]:

- La primera es el porcentaje de veces que se seleccionó una acción óptima de Pareto, que es una medida estándar en el problema del bandido.
- La segunda es una medida de pérdida (regret) propuesta específicamente para los algoritmos escalarizados. Sea el valor óptimo de la función de escalarización j:

$$f_j^* = \max_{k \in \mathcal{A}} f_j(\mu_k) \tag{7}$$

  La pérdida de elegir la acción i usando la función j es entonces:

$$\Delta_{ij} = f_j^* - f(\mu_i) \tag{8}$$

- Por último, en [4] se propone una medida de *inequidad* (unfairness). Un problema de la pérdida recién definida es que una política podría centrarse en una única solución Pareto óptima, ignorando el resto de la frontera de Pareto. Dicha estrategia podría tener una pérdida reducida, pero tardaría una enorme cantidad de tiempo en aproximar adecuadamente todas las soluciones Pareto óptimas. Por ello se propone usar adicionalmente la medida de inequidad, que no es más que la varianza del número de veces que se eligió cada accion óptima:

$$\sigma = \frac{1}{|\mathcal{A}^*|} \sum_{i \in \mathcal{A}^*} (t_i - T)^2 \tag{9}$$

  donde $\mathcal{A}^*$ es el conjunto de acciones Pareto óptimas, t_i el número de veces que se ha realizado la acción i, y T el promedio de dichos valores.

IV-B. *Resultados*

Las figuras 1, 2 y 3 muestran promedios de la probabilidad de seleccionar una acción Pareto-óptima, la pérdida acumulada, y el valor de la inequidad respectivamente, siempre en función de los pasos realizados por el agente. Se tomaron muestras de las medidas de rendimiento cada 500 pasos.

IV-C. Discusión

Como era esperable, el análisis de los resultados arroja un rendimiento muy pobre para el algoritmo C0. Al optimizarse once funciones diferentes de manera independiente, la convergencia es extremadamente lenta. Obviamente, esta tendencia dependerá fuertemente del número de funciones elegidas, lo que es un grave inconveniente para este algoritmo.

Los resultados también confirman que compartir los promedios de los valores de las acciones entre las distintas funciones acelera sensiblemente la convergencia. Concretamente, en la figura 1 se observa que C1 alcanza con unas 100.000 iteraciones el procentaje de aciertos que C0 tarda 10 veces más en alcanzar. En todos los casos, la oscilación de los valores promedio es típica para la distribución de Bernoulli utilizada en las recompensas.

Por último, la figura 1 muestra una convergencia mucho más rápida y temprana hacia acciones óptimas de C2 comparado con C0 y C1. La figura 2 confirma lo ya descrito, alcanzando C2 los menores valores de pérdida a lo largo de todo el proceso, seguido de C1 y C0.

En cuanto a la medida de inequidad, la figura 3 muestra también diferencias sensibles entre los distintos enfoques. En las fases iniciales del aprendizaje, cuando la elección de acciones es prácticamente aleatoria, es esperable que la inequidad sea reducida. Sin embargo, a medida que los algoritmos van descubriendo los óptimos de Pareto en distinto orden, es previsible también que la inequidad crezca rápidamente. Únicamente cuando se han aprendido todas las acciones Pareto-óptimas, y si el algoritmo es equitativo, la medida de inequidad comenzará a reducirse. Como es esperable, la figura 3 muestra que el algoritmo C2 comienza a desacelerar progresivamente su inequidad tras los primeros 100.000 pasos, momento en el que empieza a alcanzar una convergencia razonable. El algoritmo C1 presenta inicialmente una inequidad menor debido a su convergencia más lenta, pero acaba finalmente superando la inequidad de C2, estando con un millón de pasos todavía lejos de la deseada convergencia (vd. fig. 1). El algoritmo C0 presenta una curva casi un orden de magnitud por debajo de la de C1, pero igualmente creciente, lo cual es también coherente con la menor velocidad de convergencia ya observada en la figura 1.

En resumen, los resultados muestran que el algoritmo C2 propuesto en este trabajo puede presentar una mejora de un orden de magnitud tanto sobre C0 como sobre C1, que no es más que una mejora intuitiva de C0.

V. Conclusiones y trabajo futuro

En este trabajo se revisa el problema del bandido multiobjetivo (MOMAB) descrito originalmente en [4]. Concretamente se revisa el diseño de algoritmos basados en la escalarización de Chebyshev, que pueden ser utilizados tanto para buscar una solución específica como para aproximar toda la frontera de Pareto. Más concretamente, se propone una mejora intuitiva del algoritmo presentado en [4] (algoritmo C1), así como una reformulación del criterio de selección de acciones basado, al igual que UCB1, en acotar las estimaciones

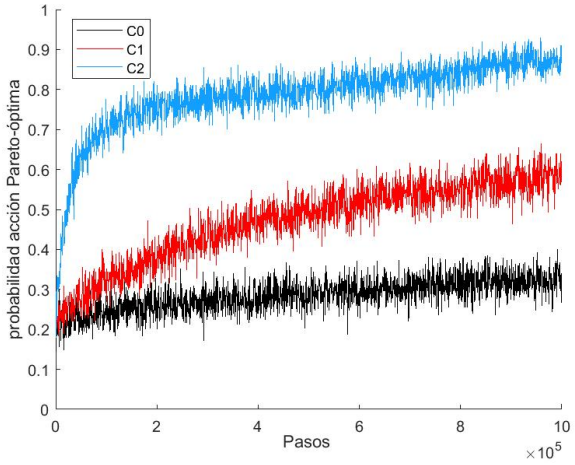


Figura 1. Probabilidad de eligir una acción Pareto-óptima.

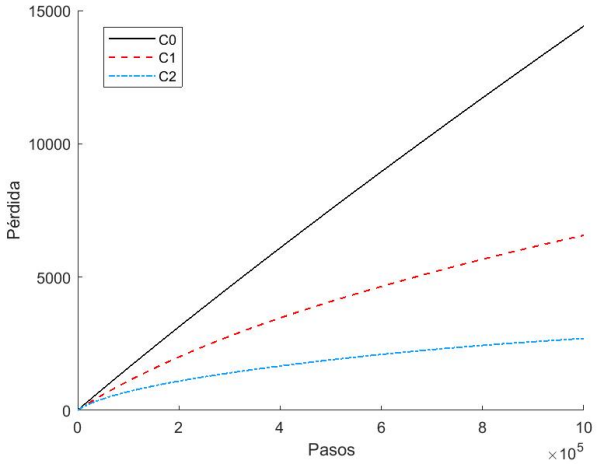


Figura 2. Pérdida promedio.

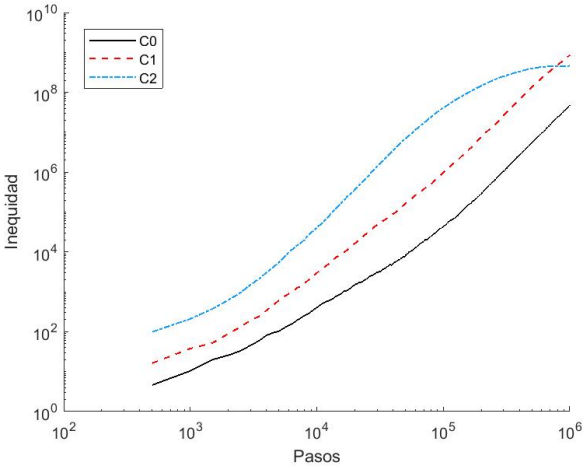


Figura 3. Inequidad promedio. Ambos ejes en escala logarítmica

de las recompensas (algoritmo C2). Se preserva así el sentido estadístico original de esta cota y de su factor de confianza. La evaluación sobre un caso de referencia muestra una mejora de C2 de al menos un orden de magnitud en la convergencia a la frontera de Pareto sobre C0 y C1. El trabajo futuro incluye una evaluación más extensa del algoritmo C2, así como su comparación con otros algoritmos multi-objetivo. También está previsto aplicar la escalarización aquí descrita a otros problemas de aprendizaje por refuerzo multi-objetivo.

Algorithm 3 Algoritmo C0. El método $ejecutar(a)$ realiza la acción a y devuelve el vector de recompensa.

Entrada N, número de pasos (decisiones); K, número de acciones; $\vec{a} = (a_1, \ldots, a_K)$, vector de acciones; S, número de funciones; $C = \{f^1, \ldots f^S\}$, conjunto de funciones (pesos).

Definir X, una matriz $K \times S$ tal que X_{ij} denota el vector promedio de recompensa obtenido para la acción i cuando la función j fue seleccionada.
Definir N, una matriz $K \times S$ tal que N_{ij} denota el número de veces que se realizó la acción i cuando la función j fue seleccionada.
Definir $\vec{t} = (t_1, \ldots t_S)$ tal que t_j denota el número de veces que la función j fue seleccionada.
for $i \in \{1..K\}$ **do**
    for $j \in \{1..S\}$ **do**
        $X_{ij} \leftarrow ejecutar(a_i)$
        $N_{ij} \leftarrow 1$
    end for
end for
for $j \in \{1..S\}$ **do**
    $t_j \leftarrow K$
end for
for cada paso en N **do**
    Elegir de manera aleatoria uniforme una función f^j
    $pasoC0(j, \vec{a}, X, N, \vec{t})$
end for

Algorithm 4 Método pasoC0.

Input j, índice de la función de escalarizado; $\vec{a}$; vector de acciones; X, N, matrices con el promedio de recompensas vectoriales, y contador de pasos; $\vec{t}$, contador de pasos globales.

$i \leftarrow \arg\max_k (f_j(X_{kj}) + \sqrt{\frac{2 \ln t_j}{N_{kj}}})$
$N_{ij} \leftarrow N_{ij} + 1$
$X_{ij} \leftarrow X_{ij} + \frac{1}{N_{ij}}(ejecutar(a_i) - X_{ij})$
$t_j \leftarrow t_j + 1$

Referencias

[1] T. Lattimore and C. Szepesvari, *Bandit algorithms*. Cambridge University Press, 2020.
[2] R. Sutton and A. Barto, *Reinforcement learning: an introduction*, 2nd ed. The MIT Press, 2018.
[3] C. Browne, E. J. Powley, D. Whitehouse, S. M. Lucas, P. I. Cowling, P. Rohlfshagen, S. Tavener, D. P. Liebana, S. Samothrakis, and S. Colton, "A survey of monte carlo tree search methods," *IEEE Trans. Comput. Intell. AI Games*, vol. 4, no. 1, pp. 1–43, 2012.

Algorithm 5 Algoritmo C1. El método $ejecutar(a)$ realiza la acción a y devuelve el vector de recompensa.

Entrada N, número de pasos; K, número de acciones; $\vec{a} = (a_1, \ldots, a_K)$, vector de acciones; S, número de funciones; $C = \{f^1, \ldots f^S\}$, conjunto de funciones.

Definir X, una matriz $K \times D$ tal que $\vec{x_i}$ denota el vector de recompensa promedio actual para la acción i (fila i).
Definir $\vec{n}$, un vector de rango K tal que n_i denota el número de veces que se realizó la acción i.
for $i \in \{1..K\}$ **do**
    $\vec{x_i} \leftarrow ejecutar(a_i)$
    $n_i \leftarrow 1$
end for
$t \leftarrow K$
for N pasos **do**
    Elegir de manera aleatoria uniforme una función f^j
    $pasoC1(j, \vec{a}, \vec{n}, X, t)$
end for

Algorithm 6 Método pasoC1.

Entrada j, índice de la función de escalarización; $\vec{a}, \vec{n}$; arrays de acciones, y contadores para cada acción; X, matriz de recompensas promedio para cada acción; t, contador de pasos global.

$i \leftarrow \arg\max_k (f^j(\vec{x_k}) + \sqrt{\frac{2 \ln t}{n_k}})$
$n_i \leftarrow n_i + 1$
$\vec{x_i} \leftarrow \vec{x_i} + \frac{1}{n_i}(ejecutar(a_i)) - \vec{x_i})$
$t \leftarrow t + 1$

[4] M. M. Drugan and A. Nowé, "Designing multi-objective multi-armed bandits algorithms: A study," in *The 2013 International Joint Conference on Neural Networks, IJCNN 2013, Dallas, TX, USA, August 4-9, 2013*. IEEE, 2013, pp. 1–8.
[5] T. Lai and H. Robbins, "Asymptotically efficient adaptive allocation rules," *Advances in Applied Mathematics*, vol. 6, no. 1, pp. 4–22, 1985.
[6] R. Agrawal, "Sample mean based index policies with o(logn) regret for the multi-armed bandit problem," *Advances in Applied Probability*, vol. 27, pp. 1054–1078, 1995.
[7] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multiarmed bandit problem," *Mach. Learn.*, vol. 47, no. 2-3, pp. 235–256, 2002.
[8] M. Ehrgott, *Multicriteria Optimization*. Springer, 2005.
[9] K. Miettinen, *Nonlinear multiobjective optimization*. Springer, 1998.

Algorithm 7 Método pasoC2.

Entrada j, índice de la función de escalarizado; $\vec{a}, \vec{n}$; arrays de acciones, y contadores para cada acción; X, matriz de recompensas promedio para cada acción; t, contador de pasos global.

$i \leftarrow \arg\max_k (f^j(\vec{x_k} + \sqrt{\frac{2 \ln t}{n_k}}))$
$n_i \leftarrow n_i + 1$
$\vec{x_i} \leftarrow \vec{x_i} + \frac{1}{n_i}(ejecutar(a_i)) - \vec{x_i})$
$t \leftarrow t + 1$

Caracterización de conjuntos de puntos mediante cierres no convexos subdivisibles

David Novoa-Paradela
CITIC, Universidad de A Coruña
A Coruña, España
david.novoa@udc.es

Oscar Fontenla-Romero
CITIC, Universidad de A Coruña
A Coruña, España
oscar.fontenla@udc.es

Bertha Guijarro-Berdiñas
CITIC, Universidad de A Coruña
A Coruña, España
berta.guijarro@udc.es

***Resumen*—Este artículo presenta un método para la caracterización de conjuntos de puntos en un plano empleando cierres no convexos subdivisibles. La solución presentada solventa la incapacidad del método de Duckham *et al.* [1] para tratar de una manera eficiente conjuntos de datos que presentan regiones separadas en el espacio. Para ello se emplean tantos cierres no convexos como sean necesarios para representar al conjunto de puntos, en lugar de un único cierre. El método mantiene la flexibilidad y eficiencia del algoritmo de Duckham *et al.*, comportándose de manera similiar ante la ausencia de regiones inconexas e incrementando su rendimiento cuando estas regiones sí que se encuentran presentes. El método puede ser de utilidad para algoritmos de aprendizaje automático en detección de anomalías y clasificación de una sola clase.**

***Palabras clave*—Cierre no convexo, Triangulación de Delaunay, Aprendizaje automático, Detección de anomalías, Clasificación de una sola clase**

I. Introducción

El cálculo de cierres convexos es una operación muy habitual en computación geométrica. Un cierre convexo o *convex hull* (CH) es el poliedro convexo más pequeño que contiene a un conjunto de puntos en el espacio. El uso de estas estructuras es habitual en problemas de detección de anomalías [2] y clasificación de una sola clase [3], [4]. Sin embargo, en muchos escenarios del mundo real la forma de los datos que estas estructuras tratan de representar no siempre será convexa. Una representación convexa nunca podrá proporcionar una buena caracterización de una distribución no convexa. Debido a esto, ha surgido toda una línea de investigación para trabajar con conjuntos de datos empleando cierres no convexos (en inglés, *non-convex hulls* o NCH). El objetivo de estas estructuras es el mismo: construir un poliedro que envuelva a un conjunto de puntos, en este caso no convexo, con suficiente precisión.

Uno de los algoritmos más flexibles y que mejores resultados obtiene construyendo cierres convexos y no convexos en espacios bidimensionales es el desarrollado por Duckham *et al.* [1]. Este método permite caracterizar a un conjunto de puntos en el plano de una manera simple y eficiciente empleando triangulaciones de Delaunay. El método logra grandes resultados, sin embargo, su rendimiento no es del todo correcto cuando se aplica sobre conjuntos de datos que presentan regiones separadas en el espacio. Estas situaciones, en las que es necesario representar más de una región aislada en el espacio, son habituales en problemas reales de detección de anomalías o clasificación de una sola clase. En estos escenarios sería conveniente disponer de más de un cierre para poder representar a cada región de manera independiente. Como se observa en la Figura 1, los resultados obtenidos por el método de Duckham *et al.* no son los más precisos en estas circunstancias ya que se emplea un único cierre para representar a todo el conjunto de puntos.

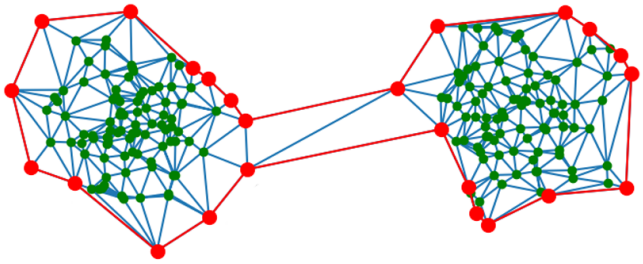


Figura 1. Cierre no convexo (rojo) resultado de aplicar el método de Duckham *et al.* sobre dos nubes de puntos separadas en el espacio.

En este trabajo presentamos un método para la caracterización de conjuntos de datos en el plano capaz de representar regiones no convexas separadas en el espacio. Para ello, partimos del método inicial de Duckham *et al.* [1] al que se añade una etapa de refinamiento de los cierres a través de un proceso de subdivisión. El método desarrollado es flexible, eficiente y solventa los problemas del método base.

II. Fundamentos teóricos

Esta sección resume las ideas principales del método de Duckham *et al*. tomado como base para el desarrollo del nuevo algoritmo.

El procedimiento desarrollado por Duckham *et al*. permite construir de una forma flexible el polígono simple que caracteriza la forma de un conjunto de puntos en el plano. Esta técnica se basa principalmente en dos tareas: realizar

Este trabajo ha sido financiado por la ayuda *Machine Learning on the Edge - Ayudas Fundación BBVA a Equipos de Investigación Científica 2019*; el Plan Nacional de Investigación e Innovación Científica y Técnica del Gobierno de España (Ayuda PID2019-109238GB-C2); y por la Xunta de Galicia (Ayuda ED431C 2018/34) con fondos FEDER de la Unión Europea. El CITIC, como Centro de Investigación del Sistema Universitario de Galicia, está financiado por la Consellería de Educación, Universidade e Formación Profesional de la Xunta de Galicia a través del Fondo Europeo de Desarrollo Regional (FEDER) y la Secretaría Xeral de Universidades (Ref. ED431G 2019/01).

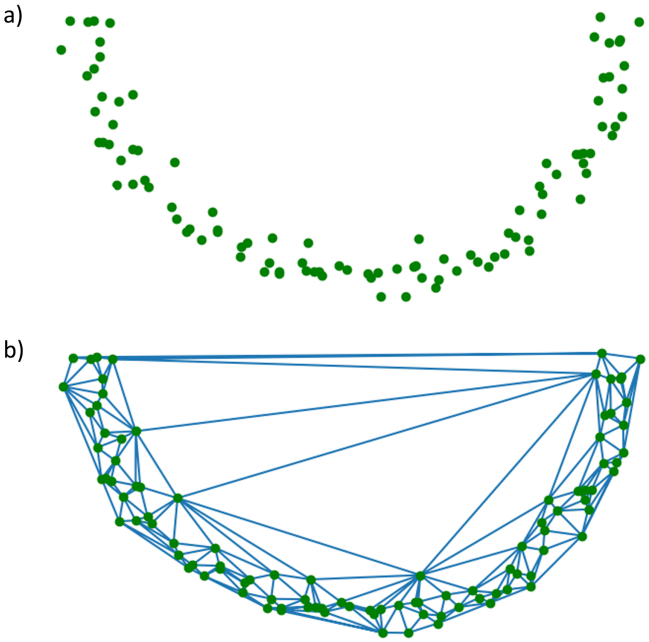


Figura 2. (a) Conjunto de datos de partida y (b) Triangulación de Delaunay obtenida.

una triangulación de Delaunay para modelar el conjunto de datos utilizando una superficie poligonal convexa, seguido de un proceso iterativo de poda para reajustar los límites de esta superficie. La superficie poligonal resultado del proceso de triangulación presenta las siguientes propiedades:

- Todos los puntos están conectados entre sí y forman tantos triángulos como sea posible sin que sus aristas se crucen.
- Los triángulos se encuentran definidos de modo que los puntos más cercanos están conectados entre sí por una arista.
- Los triángulos son lo más regulares posible, es decir, sus ángulos menores se maximizan y la longitud de sus aristas se minimiza.

En la Figura 2 se puede observar el resultado de aplicar un proceso de triangulación de Delaunay sobre un conjunto de datos con forma de media luna.

Las propiedades de la triangulación son idóneas para aplicar un proceso de poda que adapte sus límites en base a las formas no convexas de los datos. Este proceso de poda elimina iterativamente aquellas aristas exteriores del cierre que exceden un tamaño definido por el usuario mediante un parámetro l siempre que su eliminación preserve las propiedades de la triangulación. Las aristas exteriores se ordenan de mayor a menor, tras lo cual se verifica si exceden el umbral. Si una arista es mayor que l se elimina, y las dos aristas con las que formaba un triángulo pasarán a formar parte del cierre provisional. La Figura 3 muestra este procedimiento.

El proceso de poda se repetirá hasta que ninguna arista del cierre supere el tamaño definido por l. A partir de la triangulación inicial de la Figura 2, la Figura 4 muestra un

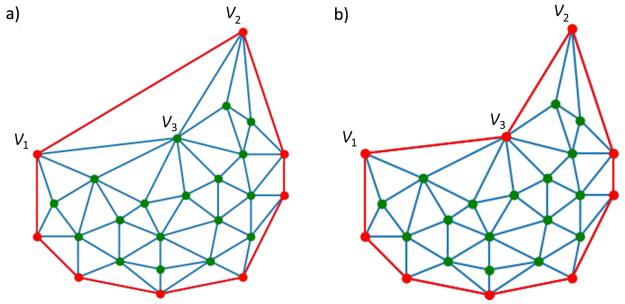


Figura 3. Proceso de poda de una arista: (a) La arista formada por los vértices $\mathbf{v}_1$ y $\mathbf{v}_2$ es la que presenta la mayor longitud del cierre (líneas rojas) y además excede el umbral establecido por el parámetro l, por lo que debe ser la primera arista en ser podada; (b) Las otras dos aristas que formaban parte del triángulo en el que se encontraba la arista podada (arista entre $\mathbf{v}_1$ y $\mathbf{v}_3$ y arista entre $\mathbf{v}_2$ y $\mathbf{v}_3$) pasarán a formar parte del cierre provisional.

ejemplo de cómo el parámetro l puede influir en el nivel de ajuste de un cierre. Como se puede ver, el conjunto de datos no es convexo, por lo que las aristas del interior de la media luna (el área no convexa) serán las primeras en podarse por ser las de mayor longitud. Un valor de l demasiado grande (1.8) podría dar como resultado cierres insuficientemente podados, mientras que un valor demasiado pequeño (0.1) podría sobreajustar los límites del cierre.

El algoritmo de Duckham *et al.* [1] es capaz de caracterizar con gran precisión tanto formas convexas como no convexas, sin embargo es incapaz de tratar con la misma precisión conjuntos de datos distribuidos en distintas regiones del espacio. Debido a este problema, proponemos un nuevo algoritmo más avanzado que permita trabajar con este tipo de escenarios.

III. Método propuesto

El objetivo principal del método desarrollado es adaptarse a conjuntos de datos que presentan regiones separadas en el espacio de manera más sofisticada que el método base de Duckham *et al.* [1], pudiendo lograr así un rendimiento superior en este tipo de escenarios. Para lograr este objetivo el método implementa una nueva fase en la que los cierres no convexos serán subdivididos de forma recursiva. Esta sección describe las etapas que componen el método.

III-A. Cálculo del cierre convexo inicial

Dado un conjunto de puntos en el plano a ser caracterizados, la primera fase del método consiste en construir un cierre convexo que envuelva al conjunto de puntos. Para llevar a cabo esta tarea se emplea el proceso de triangulación de Delaunay descrito en la sección II. El resultado de este proceso de triangulación es un cierre convexo. Si en el interior de este cierre se localizan regiones no convexas, en etapas posteriores se aplicará un proceso de poda que reajuste sus límites.

III-B. Reajuste mediante poda

Si el cierre presenta regiones no convexas en su interior se aplicará un proceso de poda que reajuste sus límites para tratar de caracterizar estas regiones no convexas. Como se

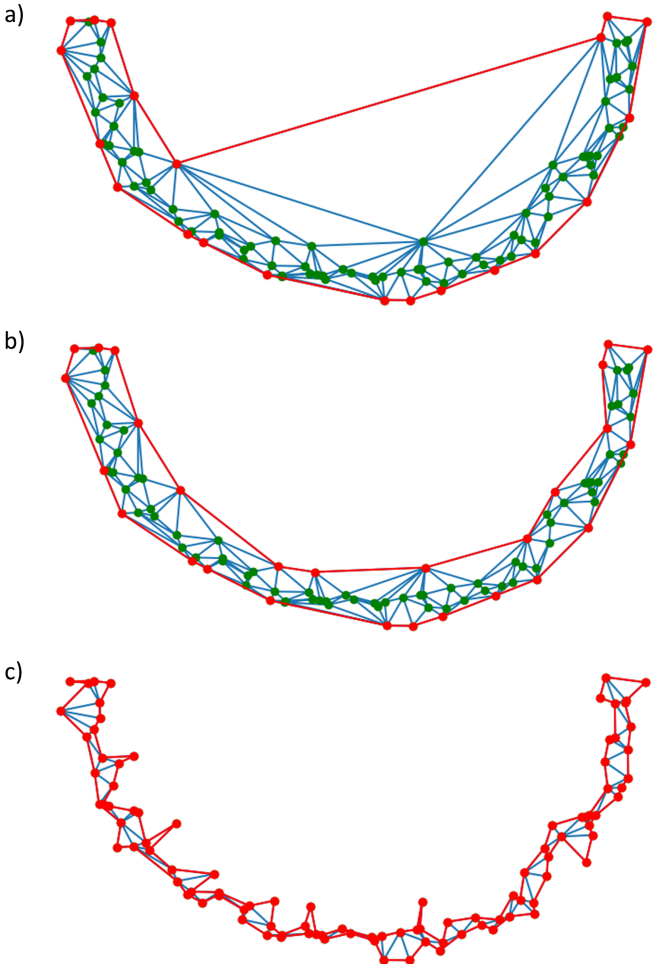


Figura 4. Cierres no convexos (rojo) resultado de aplicar el proceso de poda con distintos valores de l sobre un conjunto de datos con forma no convexa (verde) previamente triangulado (azul): (a) l = 1.8; (b) l = 0.7; (c) l = 0.1.

presentó en la sección II, las características de la triangulación permiten aplicar un proceso de poda que lleve a cabo esta tarea eliminando de forma iterativa aquellas aristas del cierre que superan un tamaño definido por el usuario mediante un parámetro l. El cierre será podado hasta que ninguna arista externa supere este tamaño.

Que este criterio de parada se cumpla no implica que el cierre se encuentre perfectamente ajustado a la forma de los datos. Como se mostraba en la Figura 1, dos regiones separadas en el espacio se encontrarán representadas erróneamente por un único cierre. Para evitar las restricciones impuestas por el uso de un único cierre, se ha implementado un proceso de subdivisión que de lugar a la coexistencia de múltiples cierres no convexos.

III-C. Subdivisión de cierres

Para diseñar el proceso de subdivisión de cierres se ha analizado el comportamiento que presenta el algoritmo de Duckham *et al.* [1] cuando es aplicado sobre conjuntos de datos que presentan regiones separadas en el espacio. Como

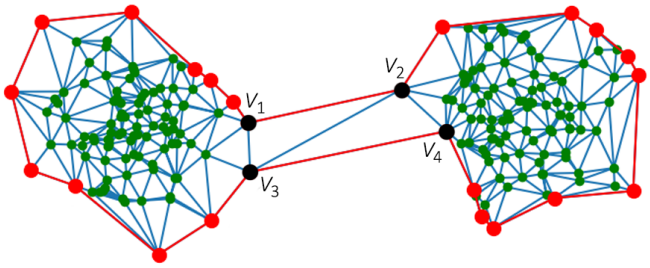


Figura 5. Morfología típica de un cierre no convexo que envuelve a dos regiones separadas en el espacio.

se puede observar en la Figura 5, si existe más de una región, el método produce un único cierre que envuelve a estas regiones separadas en el espacio manteniéndolas conectadas por una región de unión. Esta región de unión se caracteriza por la carencia de puntos en su interior, estando formada por únicamente dos triángulos cuyas aristas no fueron podadas en la etapa III-B ya que su eliminación alteraría las propiedades de la triangulación. Podemos definir a estas uniones como una región de la triangulación compuesta por dos triángulos que conforman un cuadrilátero, el cual presenta dos aristas borde opuestas y no posee puntos en su interior.

En la Figura 5, correspondiente a un conjunto de datos distribuido en dos nubes de puntos separadas en el espacio, la región de unión es la formada por los triángulos $\mathbf{v}_1$-$\mathbf{v}_2$-$\mathbf{v}_3$ y $\mathbf{v}_2$-$\mathbf{v}_3$-$\mathbf{v}_4$. Una forma intuitiva de subdividir este cierre no convexo en dos sería eliminar las dos aristas borde $\mathbf{v}_1$-$\mathbf{v}_2$ y $\mathbf{v}_3$-$\mathbf{v}_4$ y la arista común a ambos triángulos $\mathbf{v}_2$-$\mathbf{v}_3$.

Ya que la existencia de estas regiones de unión es muy recurrente tras la aplicación del método base, se ha decidido implementar un proceso de subdivisión de cierres que trate de localizar estas estructuras para tratarlas de manera específica. Para ello, las aristas que forman parte del límite del cierre serán recorridas iterativamente para comprobar la existencia de este tipo de regiones de unión. Si se determina la existencia de una unión mediante dos triángulos similar a la descrita en la Figura 5, esta será eliminada dando lugar a dos cierres no convexos independientes.

En la Figura 6 se puede observar el resultado de aplicar el método propuesto en este trabajo sobre los conjuntos de datos de la Figura 1. La utilización de varios cierres permitirá una caracterización de las regiones más precisa.

III-D. Descripción del algoritmo

El Algoritmo 1 contiene el pseudocódigo del método de caracterización de puntos en el plano presentado en este trabajo. El algoritmo comienza recibiendo como entrada un conjunto de datos bidimensionales y un valor para el parámetro l. En la línea 2 se calcula la triangulación (Δ) que contiene a los puntos del conjunto de datos, correspondiente a la etapa III-A. En las líneas 3 y 4 se crean las listas auxiliares que almacenarán la información correspondiente a las aristas y vértices de las triangulaciones (E y V). Estas estructuras serán utilizadas en fases posteriores. En la línea 5 se lleva a cabo

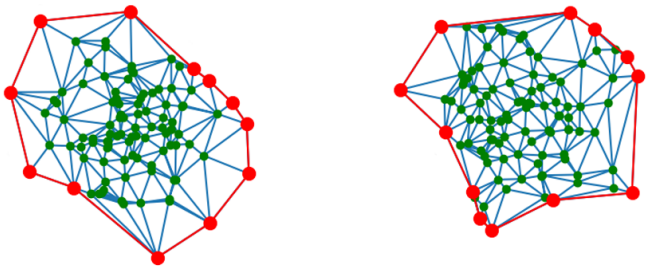


Figura 6. Cierre no convexo (rojo) resultado de aplicar el método presentado en este trabajo sobre dos nubes de puntos separadas en el espacio.

la primera fase de poda, correspondiente a la etapa III-B. En la línea 6 se lleva a cabo la primera subdivisión del cierre, correspondiente a la etapa III-C, dando lugar a la aparición de nuevas listas en E y Δ. Mientras que en la etapa de subdivisión se subdividan cierres, el algoritmo no habrá finalizado. Se repetirán las etapas de poda y subdivisión de forma iterativa hasta que no se produzcan más divisiones (líneas 7-12).

El Algoritmo 2 contiene el pseudocódigo del proceso de poda de cierres empleado por el algoritmo principal. Su funcionamiento es similar al presentado por Duckham *et al.* [1]. El algoritmo recorre las aristas borde de un cierre ordenadas de mayor a menor tamaño para eliminar aquellas que superan el valor de l definido por el usuario, preservando las características de la triangulación (líneas 2-18). Tras su ejecución, tanto la triangulación como las listas auxiliares que contienen los vértices y aristas borde (Δ, V y E) son actualizadas.

El Algoritmo 3 contiene el pseudocódigo del proceso de subdivisión de cierres empleado por el algoritmo principal. Este algoritmo recibe como entrada todos los cierres no convexos existentes, a diferencia del algoritmo 2, el cual es aplicado sobre un único cierre. El algoritmo evaluará todos los cierres que reciba como entrada para determinar si alguno de ellos es válido para ser dividido. Si detecta que algún cierre cumple los requisitos, lo dividirá en dos cierres no convexos. Se actualizará la triangulación y las listas de vértices y aristas, se detendrá la ejecución del algoritmo y se asignará el valor $True$ a la variable $seguir$, para que se repitan las etapas de poda y subdivisión en el Algoritmo 1. En caso de que ningún cierre pueda ser subdividido, $seguir$ tomará el valor $False$, lo que concluirá la ejecución del Algoritmo 1.

III-E. Complejidad del algoritmo

La complejidad del algoritmo de Duckham *et al.* es según los autores $O(n \log n)$, siendo n el número de puntos a caracterizar [1]. Nuestra versión del algoritmo añade una nueva etapa de subdivisión cuyo objetivo principal es localizar las regiones de unión para dividir los cierres. El algoritmo itera sobre cada arista externa hasta localizar una que podría pertenecer a una región de unión. Cuando esta primera arista es localizada, el algoritmo vuelve a iterar sobre las aristas externas, buscando una segunda arista con la que formaría dicha estructura de unión. Este proceso de búsqueda se repite hasta que se encuentra una región de unión válida o se han evaluado todas las aristas externas sin éxito. El número de aristas externas será el mismo que el de vértices externos, y este, en la mayor parte de los casos, será mucho menor que n, ya que solo un subconjunto de los puntos serán vértices externos. Es por ello que para esta tarea de subdivisión estimamos una complejidad $O(v^2)$, siendo v el número de vértices externos. Las operaciones asociadas a la subdivisión de los cierres, como la eliminación y creación de aristas, pueden ser abordadas en tiempos constantes, por lo que la complejidad final del algoritmo será $O(n \log n + v^2)$. La complejidad se ha visto incrementada debido a la incorporación de una nueva etapa, sin embargo en la práctica el número de vértices v es muy inferior al número total de puntos n, lo que reduce el impacto de este incremento en la complejidad.

Algorithm 1 : Algoritmo principal para la caracterización de un conjunto de puntos en el plano mediante cierres no convexos subdivisibles.

Inputs:
$\mathbf{X} \in \mathbb{R}^{2\times n}$ ▷ Dataset (2 variables $\times$ n instancias)
$l \in \mathbb{R}^{+}$ ▷ Tamaño máximo de arista permitida
Output:
E ▷ Lista de cierres no convexos

1: **function** COMPUTENCH
2: Δ[1] = triangulación de Delaunay sobre **X**
3: E[1] = lista de aristas borde de Δ
4: V[1] = lista de vértices borde de Δ
5: E[1], V[1], Δ[1] = Podado(E[1], V[1], Δ[1], l)
6: E, Δ, $seguir$ = Subdivision(E, Δ, l, X)
7: **while** $seguir == True$ **do**
8: **for** $i = 1..len(E)$ **do** ▷ Para cada cierre
9: E[i], V[i], Δ[i] = Podado(E[i], V[i], Δ[i], l)
10: **end for**
11: E, Δ, $seguir$ = Subdivision(E, Δ, l)
12: **end while**
13: **return** E
14: **end function**

IV. RESULTADOS

La mejora en la capacidad de adaptación del método ante regiones separadas en el espacio es apreciable de una forma intuitiva a través de pruebas gráficas como las Figuras 7 y 8. Para ello, la utilización de conjuntos de datos artificiales con formas características (como medias lunas o circunferencias separadas en el espacio), nos permite evaluar cualitativamente la capacidad de los métodos, ya que podemos comparar los límites generados por estos con los que idóneamente esperaríamos para cada una de las formas representadas.

Desde el punto de vista cualitativo, está claro que las implementación de las mejoras presentadas en la sección III han dotado al método de la capacidad de subdivisión y adaptación ante conjuntos de datos en los que existen regiones inconexas.

Algorithm 2 : Función auxiliar empleada por el método principal para llevar a cabo el proceso de poda de las aristas de un cierre.

Inputs:
- E_i ▷ Cierre no convexo
- V_i ▷ Vértices borde del cierre
- Δ_i ▷ Triangulación del cierre
- l ▷ Tamaño máximo de las aristas

Output:
- FB ▷ Nuevas aristas borde del cierre
- V_i ▷ Nuevos vértices borde del cierre
- Δ_i ▷ Nueva triangulación

```
function PODADO
    FB = ∅   ▷ Lista para almacenar a las aristas finales
    Ordena la lista E_i de mayor a menor tamaño
    Ordena la lista V_i por vértices consecutivos
    while E_i no esté vacía do
        Asigna e ← head(E_i)
        Elimina e de E_i
        if ‖e‖ > l then
            Busca el triángulo t de Δ_i que contiene a los dos vértices de e y obtiene su tercer vértice v
            if not (v in V_i) then
                Elimina e de la triangulación Δ
                Inserta las otras dos aristas de t en E_i ordenadamente
                Inserta v en V_i ordenadamente
            end if
        else
            Inserta e en FB
        end if
    end while
    return FB, V_i, Δ_i
end function
```

Algorithm 3 : Función auxiliar empleada por el método principal para llevar a cabo el proceso de subdivisión de cierres.

Inputs:
- E ▷ Lista de cierres no convexos
- Δ ▷ Lista de triangulaciones, una por cierre
- l ▷ Tamaño máximo de las aristas

Output:
- E ▷ Lista de cierres no convexos actualizada
- Δ ▷ Lista de triangulaciones actualizada
- $seguir$ ▷ Valor $True$ si se ha producido alguna subdivisión o $False$ en caso negativo.

```
function SUBDIVISION
    seguir = False
    for NCH, T in E, Δ do ▷ Para cada NCH
        for e_1 in NCH do ▷ Para cada arista de NCH
            if ‖e_1‖ > l then
                Obtén el triángulo t_1 de T que contiene a los dos vértices de e_1
                Obtén el tercer vértice v_1 de t_1
                Crea la lista e_list de aristas en las que se encuentra v_1
                for e_2 in e_list do
                    Crea la lista t_list de triángulos a los que pertenece e_2
                    for t_2 in t_list do
                        Obtén el tercer vértice v_2 de t_2
                        if v_2 ∈ e_1 then
                            Elimina e_1 de NCH y de T
                            Elimina e_2 de NCH y de T
                            Elimina arista v_1-v_2 de T
                            Añadir nuevas aristas borde a E
                            Divide NCH y actualiza E
                            Divide T y actualiza Δ
                            seguir = True
                            Break ▷ Finalizar subdivisión
                        end if
                    end for
                end for
            end if
        end for
    end for
    return E, Δ, seguir
end function
```

Para disponer además de una evaluación cuantitativa del progreso del método, se ha decidido estudiar el impacto que produce la subdivisión de los cierres en el área que estos presentan. Si el método de Duckham *et al.* [1] y el nuevo método se aplican sobre conjuntos de datos que presentan regiones separadas en el espacio, el primero empleará un único cierre no convexo, mientras que el segundo llevará a cabo procesos de subdivisión y poda que supondrán la creación de varios subcierres no convexos. El área total de estos subcierres (la suma de las áreas de cada subcierre) será menor que el área total del único cierre generado por el método de Duckham *et al.* Además, idealmente, si ambos métodos se aplican sobre un conjunto de datos que no presenta regiones separadas en el espacio, es decir, representable mediante un único cierre no convexo, los cierres obtenidos por ambos métodos deben ser idénticos, y con ello, su área.

Para llevar a cabo este estudio se ha comparado el área de los cierres obtenidos por el método base y el método presentado en este artículo sobre diferentes conjuntos de datos generados artificialmente [5]. La Tabla I recoge la diferencia entre las áreas de los cierres generados por los dos métodos. Como se puede observar, la utilización de más de un cierre sobre regiones separadas en el espacio, como por ejemplo dos circunferencias, permite reducir el área de los cierres finales, producto de un ajuste más preciso a la forma de los datos. Esta reducción del área necesaria para caracterizar a los datos es notable en algunos conjuntos tales como las cuatro circunferencias o las dos medias lunas enfrentadas.

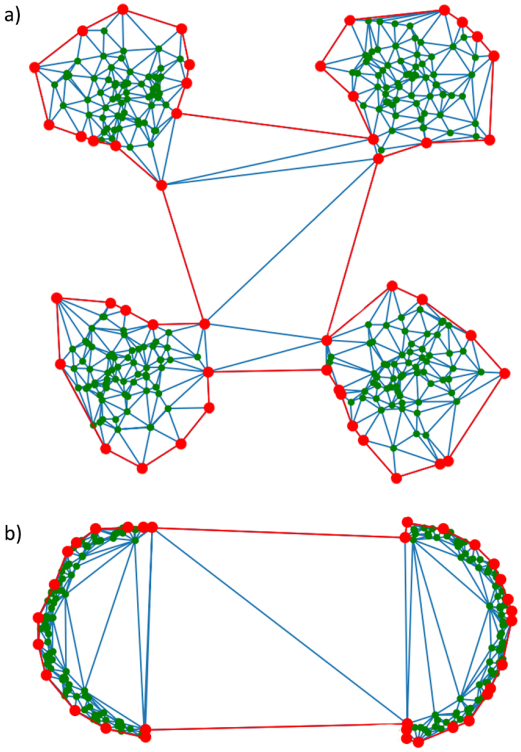


Figura 7. Cierres no convexos (rojo) resultado de aplicar el método de Duckham *et al.* sobre diferentes conjuntos de datos (verde) previamente triangulados (azul): (a) Cuatro circunferencias separadas en el espacio; (b) Dos medias lunas enfrentadas.

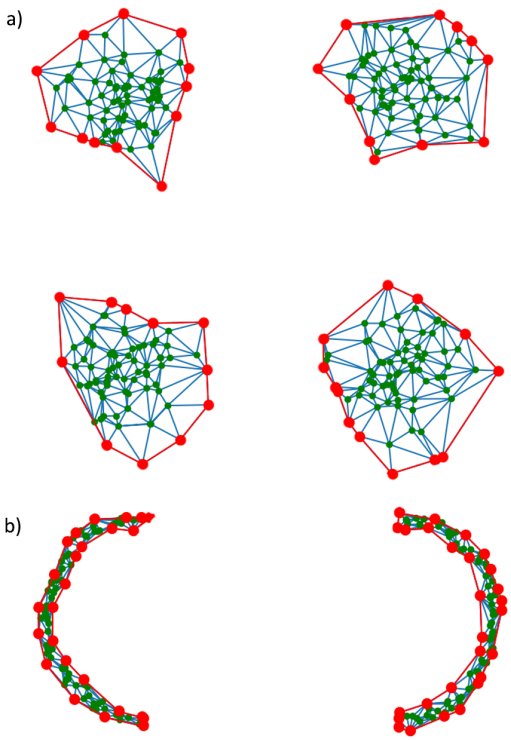


Figura 8. Cierres no convexos (rojo) resultado de aplicar el método presentado en este trabajo sobre diferentes conjuntos de datos (verde) previamente triangulados (azul): (a) Cuatro circunferencias separadas en el espacio; (b) Dos medias lunas enfrentadas.

Tabla I
REDUCCIÓN DEL ÁREA DE LOS CIERRES PRODUCIDOS POR EL MÉTODO PRESENTADO EN ESTE TRABAJO FRENTE AL MÉTODO DE DUCKHAM *et al.* PARA DISTINTOS CONJUNTOS DE DATOS ARTIFICIALES.

Conjunto de datos	Diferencia de área
Una circunferencia	0.0 %
Dos circunferencias	-11.0 %
Tres circunferencias	-17.8 %
Cuatro circunferencias	-37.2 %
Una media luna	0.0 %
Dos medias lunas enfrentadas	-86.1 %
Dos medias lunas entrelazadas	-2.8 %
Dos anillos concéntricos	0 %

Se ha confirmado además que, cuando el nuevo método es aplicado sobre conjuntos de datos representables mediante un único cierre, este no lleva a cabo subdivisiones innecesarias, manteniendo el comportamiento del método base. Esto se puede observar en el caso de los conjuntos de datos con forma de media luna o de una única circunferencia, donde los cierres finales de ambas aproximaciones son iguales, y con ello sus áreas.

V. CONCLUSIONES

La implementación de los procesos recursivos de subdivisión y poda han dotado al método de la capacidad de adaptación ante conjuntos de datos en los que existen regiones inconexas. El método mantiene la flexibilidad y eficiencia del algoritmo Duckham *et al.* [1] comportándose de manera similar ante conjuntos de datos que no presentan regiones separadas en el espacio.

Las regiones que el método Duckham *et al.* era incapaz de eliminar del cierre no convexo final están caracterizadas por la inexistencia de puntos en su interior. La expulsión de estas regiones vacías facilita la aplicación del nuevo método en campos como la detección de anomalías o la clasificación de una sola clase, tareas cuyo objetivo es la caracterización de un único conjunto de datos con la mayor precisión posible. En problemas de este tipo es posible que los conjuntos de datos a modelar presenten regiones separadas en el espacio, por lo que la eliminacion de las regiones de unión vacías permitiría reducir el número de clasificaciones erróneas (tasa de falsos negativos en el caso de la detección de anomalías).

REFERENCIAS

[1] M. Duckham, L. Kulik, M. Worboys, and A. Galton, "Efficient generation of simple polygons for characterizing the shape of a set of points in the plane," *Pattern Recognition*, vol. 41, no. 10, pp. 3224–3236, 2008.
[2] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Computing Surveys*, vol. 41, no. 3, pp. 15:1–15:58, 2009.
[3] P. Perera, P. Oza, and V. M. Patel, "One-class classification: A survey," 2021.
[4] S. S. Khan and M. G. Madden, "One-class classification: Taxonomy of study and review of techniques," *The Knowledge Engineering Review*, vol. abs/1312.0049, 2013.
[5] S. Learn. (2016) Artificial datasets. https://scikit-learn.org/stable/datasets.html.

A memetic algorithm to minimize the total weighted tardiness in the fuzzy flexible job shop

Pablo García Gómez
University of Cantabria
Santander, Spain

Camino R. Vela
University of Oviedo
Oviedo, Spain

Inés González-Rodríguez
University of Cantabria
Santander, Spain

***Abstract*—This paper considers the flexible job shop scheduling problem with uncertainty in the processing time of operations minimizing the total weighted tardiness. We generalize the different categories of schedules existing in the deterministic job shop to the case of fuzziness and flexibility, based on this, we propose a memetic algorithm, a hybrid between an evolutionary algorithm and tabu search. We adapt a crossover operator from the traditional job shop to handle flexibility and we improve a recently published neighbor filtering mechanism to reduce its computational complexity while maintaining its effectiveness. An experimental study confirms a synergy effect between both the evolutionary and the tabu search components of the memetic algorithm. It also shows the potential of the proposed method, with results comparable to the state-of-the-art in terms of the more extended makespan objective function. For the total weighted tardiness there are no previous results in the literature so we provide a first set of results as a reference for future research.**

***Index Terms*—flexible job shop scheduling, due dates, total weighted tardiness, fuzzy numbers, memetic algorithms, evolutionary algorithms, tabu search.**

I. Introduction

Scheduling is a critical problem that humans have to solve every day. Some examples are organizing the manufacturing process in industrial systems, deciding the order in which tasks have to be executed in a CPU or controlling air traffic in an airport [1].

The job shop scheduling problem (JSP) tries to model in a generic way those problems that consist in planning the execution of a series of operations on a finite set of resources. In this work, we add two extensions to the traditional problem to bring it closer to real-life problems. The first one is flexibility, which allows an operation to be executed in different machines. The second one is uncertainty in the processing time of operations, modeled using triangular fuzzy numbers. The resulting problem can be referred to as the fuzzy flexible job shop scheduling problem. For this variant, we generalize for the first time the categories of schedules defined in the literature for the traditional problem.

The objectives in a JSP can be diverse. Historically, the main objective has been minimizing the makespan, i.e., the maximum completion time. However, in recent years, attention has shifted to meeting due dates as a response to the changes in production. For this reason, we consider minimizing the total weighted tardiness, which measures the delay of the jobs with respect to their due dates, taking into account that some jobs may be more important than others.

The JSP is an NP-Hard problem [2] and therefore intractable with exact algorithms. In addition, the total weighted tardiness has a higher computational complexity than the more traditional makespan and the added flexibility and uncertainty further increase the difficulty since solving methods must consider more variables in order to obtain a solution. Thus, the problem is a perfect candidate to be solved with metaheuristic search methods.

Indeed, several metaheuristic methods have been proposed for the job shop problem with flexibility and fuzziness, either for makespan minimization or in multiobjective settings (cf. [3] and [4] and references therein). However, to the best of our knowledge, the total weighted tardiness objective has never been considered in the fuzzy flexible variant of the problem.

We propose a memetic algorithm similar to the one used in [3] for makespan, which combines an evolutionary method with tabu search. In this work we propose two novel elements. On the one hand, we have adapted a crossover operator that has shown good performance for the traditional JSP to the case of flexibility with the objective of better combining the characteristics of the solutions. On the other hand, to improve performance, we have modified a neighbor filtering mechanism from [5], reducing its computational complexity without affecting its behavior.

The rest of the paper is organized as follows. Section II formally defines the problem. Section III describes the memetic algorithm. In Section IV we report and make an analysis of the obtained experimental results to evaluate the potential of our proposal. Finally, Section V presents the conclusions.

II. Problem Formulation

The job shop scheduling problem consists in scheduling a set of jobs $\mathcal{J} = \{J_1, \ldots, J_n\}$ in a set of machines $\mathcal{M} = \{M_1, \ldots, M_m\}$ subject to a set of constraints. Each job J_j consists of n_j operations (also called tasks) that must be sequentially scheduled, i.e., there are precedence constraints between them. An operation may be denoted o_{jk} where j denotes the job J_j to which it belongs and k denotes the position in the job in which it must be executed. The set of all operations will be denoted as $\mathcal{O}$. There are also

This research has been supported by Grant Concepción Arenal UC-20-20 from the University of Cantabria and the Government of Cantabria and Grant PID2019-106263RB-I00 from the Government of Spain.

capacity constraints, by which each operation requires the uninterrupted and exclusive use of one of the machines for its whole processing time. When flexibility is considered, each operation o_{jk} may be executed in any machine from a given set $\mathcal{M}_{jk} \subseteq \mathcal{M}$ and its processing time $p_{o_{jk}\mu}$ depends on the machine $\mu \in \mathcal{M}_{jk}$ where it is executed. Additionally, each job has a due date d_j by which it should be completed and a weight w_j indicating its priority.

A solution $(\boldsymbol{\mu}, \mathbf{s})$ to the problem (also called schedule) consists of both a machine assignment $\boldsymbol{\mu}$ and starting time assignment $\mathbf{s}$ to all operations. A solution is said to be feasible if all constraints hold. More formally, let μ_{jk} be the machine assigned to operation o_{jk} and s_{jk} and $c_{jk} = s_{jk} + p_{o_{jk}\mu_{jk}}$ be its starting and completion times respectively. Then, precedence constraints hold if $c_{jk} \leq s_{j(k+1)}$ for $1 \leq j \leq n, 1 \leq k < n_j$ and capacity constraints hold if $c_{jk} \leq s_{il} \vee c_{il} \leq s_{jk}$ when $\mu_{jk} = \mu_{il}$ for $1 \leq i \leq n, 1 \leq j \leq n, 1 \leq k \leq n_j, 1 \leq l \leq n_i$. Given this definition, the starting time assignment induces a linear processing order $\boldsymbol{\sigma}$ and a processing order in each machine $\boldsymbol{\delta}_\mu$. Consistently with the rest of the notation used, σ_{jk} represents the position of operation o_{jk} in $\boldsymbol{\sigma}$.

For a feasible solution, the total weighted tardiness is defined as $TWT = \sum_{j=1}^{n} w_j T_j$ where $T_j = \max(0, C_j - d_j)$ is the tardiness of job J_j with respect to its due date d_j and $C_j = c_{jn_j}$ denotes its completion time. Notice that TWT does not penalize jobs that finish before their due date, only those that finish after it.

A. Fuzzy Scheduling

It is difficult to know beforehand how long an operation will take, instead, we usually know a range of possible values along with the most likely one. We introduce this information in the problem by modeling processing times as triangular fuzzy numbers, a particular type of fuzzy numbers [6].

Definition 1. *Triangular Fuzzy Number.* A triangular fuzzy number (TFN) is a fuzzy number with an interval $[a^1, a^3]$ of possible values and a modal value a^2. A TFN can be represented as $a = (a^1, a^2, a^3)$ and its membership function is given by the following expression:

$$\mu_a(x) = \begin{cases} \frac{x-a^1}{a^2-a^1} & a^1 \leq x \leq a^2 \\ \frac{x-a^3}{a^2-a^3} & a^2 \leq x \leq a^3 \\ 0 & x < a^1 \vee a^3 < x \end{cases}$$

For our problem, we need three arithmetic operations, the sum of TFNs, the maximum of TFNs and the subtraction between a TFN and a crisp number (because we consider due dates as crisp numbers). All these operations can be obtained using the extension principle [7]. This way the sum of two TFNs a and b is defined as $a + b = (a^1 + b^1, a^2 + b^2, a^3 + b^3)$ and the subtraction of a TFN with a crisp number c is defined as $a - c = (a^1 - c, a^2 - c, a^3 - c)$. Unfortunately, the set of TFNs is not closed under the maximum and its computation can result cumbersome. For this reason, it is usual to approximate the maximum between two TFNs as $\max(a, b) \approx (\max(a^1, b^1), \max(a^2, b^2), \max(a^3, b^3))$

There exists no natural relation of total order for TFNs. Thus, to compare solutions based on their TWT it is common in the literature to turn to some ranking method. Here, we will use a ranking based on the expected value (also the neutral scalar substitute) of TFNs, so $a \leq_E b$ iff $E[a] \leq E[b]$, where $E[a] = \frac{a^1 + 2a^2 + a^3}{4}$.

Additionally, we introduce two dominance relations between TFNs. We shall say that a TFN a strongly dominates TFN b, denoted $a \succ_D b$, when $a^1 > b^1 \wedge a^2 > b^2 \wedge a^3 > b^3$. Similarly, a weakly dominates b, denoted $a \succ_d b$, when $(a^1 \geq b^1 \wedge a^2 \geq b^2 \wedge a^3 \geq b^3) \wedge (a^1 > b^1 \vee a^2 > b^2 \vee a^3 > b^3)$.

These relations allow us to extend some well-known concepts from deterministic scheduling to the fuzzy framework by defining different categories of fuzzy schedules.

Definition 2. *Semi-active schedule.* A fuzzy schedule $(\boldsymbol{\mu}, \mathbf{s})$ is said to be semi-active if no operation can start earlier without changing the processing order of any machine, that is:

$$\nexists\, \mathbf{s}' : \exists\, o_{jk} : s'_{jk} \prec_d s_{jk} \wedge s'_{lm} = s_{lm} \forall o_{lm} \in \mathcal{O} \setminus \{o_{jk}\}$$
$$\wedge\, \boldsymbol{\delta}'_\mu = \boldsymbol{\delta}_\mu \forall \mu \in \mathcal{M}$$

Definition 3. *Active schedule.* A fuzzy schedule $(\boldsymbol{\mu}, \mathbf{s})$ is said to be active if no operation can start earlier without delaying any other operation, that is:

$$\nexists\, \mathbf{s}' : \exists\, o_{jk} : s'_{jk} \prec_d s_{jk} \wedge s'_{lm} = s_{lm} \forall o_{lm} \in \mathcal{O} \setminus \{o_{jk}\}$$

Given these definitions, the class of semi-active schedules strictly includes the class of active schedules. Most importantly, it can be proved that there exists at least an optimal solution in the class of active schedules. This allows restricting the search space to the class of semi-active or active solutions, as deemed more appropriate.

In this work we will only consider semi-active solutions, for this reason, we will also call solution to the machine assignments plus the linear order of the operations $(\boldsymbol{\mu}, \boldsymbol{\sigma})$ because from this order starting times can be inferred assigning to each operation its earliest possible starting time.

III. Memetic Algorithm

As the solving method, we propose a memetic algorithm, a hybrid algorithm that combines an evolutionary algorithm with local search (tabu search in our case), so the individuals from the evolutionary algorithm act as initial solutions for the local search. The objective is to benefit from the synergies between both methods, finding an equilibrium between exploration and exploitation.

A. Evolutionary Component

Looking for diversity, the algorithm starts from a randomly generated population. Then, in each iteration, the whole population is randomly organized into couples, giving every individual a chance to reproduce. These couples are combined using a crossover operator in order to obtain two descendants. Finally, a new population is built by means of a tournament such that the best two from each pair of parents and their two

Algorithm 1: G&T.

Input: $(\boldsymbol{\mu}, \boldsymbol{\sigma})$
Output: $(\boldsymbol{\mu}, \mathbf{s})$ ▷ s may induce a processing order different than $\boldsymbol{\sigma}$
L ; ▷ Stores the last operation scheduled in each machine
$A = \{o_{11}, o_{21}, \ldots, o_{n1}\}$; ▷ Initial operations of the jobs
while $A \neq \emptyset$ **do**
  ▷ Select the operation that can be completed earlier
  $o_{il} = argmin_{\leq_E}(max(c_{jk-1}, c_{L[\mu_{jk}]}) + p_{o_{jk}\mu_{jk}} : o_{jk} \in A)$;
  ▷ Select all operations that can start earlier in the same machine (conflict set)
  $t = max(c_{il-1}, c_{L[\mu_{il}]}) + p_{o_{il}\mu_{il}}$;
  $E = \{o_{jk} : o_{jk} \in A \wedge \mu_{jk} = \mu_{il} \wedge max(c_{jk-1}, c_{L[\mu_{jk}]}) <_E t\}$;
  ▷ Select the highest priority operation
  $o_{il} = argmin_{\leq_E}(\sigma_{jk} : o_{jk} \in E)$;
  $s_{il} = max(c_{il-1}, c_{L[\mu_{il}]})$;
  $c_{il} = s_{il} + p_{o_{il}\mu_{il}}$;
  $L[\mu_{il}] = o_{il}$;
  $A = A \setminus \{o_{il}\}$;
  ▷ Add the successor in the job (if it exists)
  $A = A \cup \{o_{il+1}\}$;
end
return $(\boldsymbol{\mu}, \mathbf{s})$

offspring are chosen. The process is repeated until a stopping criterion is met.

Since we restrict to the set of semi-active schedules, we represent a solution as $(\boldsymbol{\mu}, \boldsymbol{\sigma})$. However, to reduce the size of the search space and compensate the bias of evolutionary algorithms towards exploration, $(\boldsymbol{\mu}, \boldsymbol{\sigma})$ will be decoded into an active schedule. To do so, a modified version of the G&T algorithm [8] is proposed in Algorithm 1. It can be proved that this version of the algorithm is not complete. However, this is not that relevant in the context of metaheuristic search, especially since the produced solution will be used as a (hopefully good) starting point for a local search procedure. In the process of building the active schedule, it may be necessary to alter the linear order of operations. In that case, Lamarckism is applied and the original $\boldsymbol{\sigma}$ is changed accordingly.

Our crossover operator is an extension to the case of flexibility of the Generalized Order Crossover (GOX) [9]. As far as we know such an extension has not been proposed in any previous work.

In order to obtain an offspring, one of the parents is designated as the receptor and the other as the donor. Exchanging roles a second offspring can be obtained. Then we randomly select a subsequence of the donor's $\boldsymbol{\sigma}$, called implant. The length of this subsequence should be between one-half and one-third of the total length. We distinguish two cases. If the implant lies inside the donor, it is inserted in the receptor's $\boldsymbol{\sigma}$ right after the operation that equals the first operation of the implant. On the other hand, if the implant is wrapped around, it is inserted at each end of the receptor's $\boldsymbol{\sigma}$. Notice that when the implant lies inside the donor, the resulting $\boldsymbol{\sigma}$ might be unfeasible because precedence constraints are not necessarily satisfied. To solve this, all operations of a job J_j are reordered using the positions assigned to J_j in $\boldsymbol{\sigma}$ so that the constraints hold. This can be considered an implicit mutation but the chosen insertion point ensures that it rarely happens. Finally, and common to both cases, to build $\boldsymbol{\mu}$ each operation is assigned the same machine as in the parent it comes from.

B. Local Search Component

Individuals generated at each iteration of the evolutionary algorithm will be improved using tabu search. Tabu search [10] is a local search algorithm whose main defining characteristic is that it keeps a memory structure, called tabu list. This structure stores a trace of the recently visited search space with the objective of avoiding getting stuck in local optima. In particular, we store in the tabu list the inverse of the move that is performed in each iteration to go from one solution to a neighbor. This way, the search avoids undoing recently made moves. We use a dynamic size tabu list, with a resizing mechanism similar to the one introduced in [11], so the size of the list can vary between a lower bound and an upper bound. When the selected neighbor is worse (resp. better) than the current solution, the list's size increases (resp. decreases) in one unit, unless the upper (resp. lower) bound has been reached. If the selected neighbor is the best solution found so far, the list is cleared; it is similar to restarting the search from this solution. We also incorporate an aspiration criterion so a tabu move can be executed if it improves the best solution found up to this moment. In the rare situation that all neighbors are tabu, we choose the best one and clear the tabu list.

Now we will define the key component of any local search, the neighborhood function. This function relies on the concept of criticality, extended here to the fuzzy flexible JSP minimizing the TWT in a similar fashion to the case for the makespan in [3]. For a solution $\phi = (\boldsymbol{\mu}, \boldsymbol{\sigma})$ and an operation o, let PJ_o (resp. SJ_o) denote its predecessor (resp. successor) in its job, $PM_o(\phi)$ (resp. $SM_o(\phi)$) its predecessor (resp. successor) in its machine and $p_o(\phi)$ its processing time in the machine it is assigned to.

Definition 4. *Head.* The head $r_o(\phi)$ of an operation o in a solution ϕ is equal to its earliest starting time, that is:

$$r_o(\phi) = \max\{r_{PJ_o}(\phi) + p_{PJ_o}(\phi), r_{PM_{o(\phi)}}(\phi) + p_{PM_{o(\phi)}}(\phi)\}$$

with $r_o(\phi) = (0, 0, 0)$ if o has no predecessors.

Definition 5. *Tail.* The tail $q_{oj}(\phi)$ of an operation o and a job J_j in a solution ϕ is the time left after o has been processed until the job J_j is completed, that is:

$$q_{oj}(\phi) = \max\{q_{SJ_oj}(\phi) + p_{SJ_o}(\phi), q_{SM_{o(\phi)}j}(\phi) + p_{SM_{o(\phi)}}(\phi)\}$$

with $q_{oj}(\phi) = (0, 0, 0)$ if o has no successors and belongs to job J_j, and $q_{oj}(\phi) = (-Q, -Q, -Q)$ if o has no successors and does not belong to job J_j where Q is an arbitrarily large number ($Q > \max_{o \in \mathcal{O}} r_o(\phi) + p_o(\phi)$)

Heads (resp. tails) must be computed in such a way that the predecessors (resp. successors) have already been evaluated.

Definition 6. *Critical operation.* An operation o is said to be critical for a job J_j in a solution ϕ if there exists a component $i \in \{1,2,3\}$ such that $T_j^i = \max((0,0,0), r_o(\phi) + p_o(\phi) + q_{oj}(\phi) - d_j)^i$

Definition 7. *Critical block.* A critical block for a job J_j and a component i of the fuzzy TWT is a maximal sequence B of tasks all requiring the same machine, such that no pair of consecutive tasks belong to the same job and where every task $o \in B$ is critical for job J_j and $T_j^i = \max((0,0,0), r_o(\phi) + p_o(\phi) + q_{oj}(\phi) - d_j)^i$

Clearly, a neighbor can only improve in terms of TWT if there is some change in critical tasks. Also, if a neighbor is obtained by exchanging the position of two consecutive operations, it can be proved that it can only improve in terms of TWT if the operations lie at the extreme of a critical block.

We will consider a neighborhood that is the union of two smaller neighborhoods, one that acts on the assigned machines (N_{MR}) based on [12] and one that acts on the order of the operations (N_{CET}) based on [13].

Definition 8. *Machine Reassignment Neighborhood (N_{MR}).* Let $\phi = (\boldsymbol{\mu}, \boldsymbol{\sigma})$ be a feasible solution. Let $\boldsymbol{\mu}_{(o_{jk},a)}$ denote the assignment that results from reassigning operation o_{jk} to machine M_a. Then $N_{MR}(\phi) = \{(\boldsymbol{\mu}_{(o_{jk},a)}, \boldsymbol{\sigma}) : o_{jk}$ is critical, $M_a \in \mathcal{M}_{jk}, M_a \neq \mu_{jk}\}$

Definition 9. *Critical End Transpose Neighborhood (N_{CET}).* Let $\phi = (\boldsymbol{\mu}, \boldsymbol{\sigma})$ be a feasible solution. Let $\boldsymbol{\sigma}_{(u,v)}$ denote the operation processing order that results from inverting the positions of operations u and v in the same machine. Then $N_{CET}(\phi) = \{(\boldsymbol{\mu}, \boldsymbol{\sigma}_{(u,v)}) : (u,v)$ is the first or last arc of a critical block$\}$

It can be demonstrated that the solutions generated by both neighborhoods are feasible.

Observe that N_{MR} dominates N_{CET} in size and therefore in computational complexity. Indeed, if $t = |\mathcal{O}|$ is the total number of operations, then the size of N_{CET} in the worst case is $2t/3$ and the size of N_{MR} in the worst case is $t(m-1)$. Thus, the complete neighborhood contains at most $2t/3 + t(m-1)$ solutions, which in asymptotic notation can be expressed as $\mathcal{O}(tm)$.

Evaluating neighbors can be heavily time-consuming, for this reason, several ways of approximating the quality of a solution can be found in the literature. One of the most extended ones for the makespan is proposed in [14]. This approximation can be adapted to TWT as follows.

From the definition of heads and tails and given a feasible solution ϕ, we can calculate a lower bound (a value weakly dominated by or equal to the real value) for the completion time C_j of job J_j in a neighbor $\phi' \in N_{MR}(\phi)$ and in a neighbor $\phi'' \in N_{CET}(\phi)$ as:

$$LB(C_j(\phi')) = \begin{cases} r_o(\phi') + p_o(\phi') + q_{oj}(\phi') \\ \qquad \text{if } (0,0,0) \not\succ_D q_{oj}(\phi) \\ C_j(\phi) \qquad \text{otherwise} \end{cases}$$

$$LB(C_j(\phi'')) = \begin{cases} \max\{r_u(\phi'') + p_u(\phi'') + q_{uj}(\phi''), \\ \qquad r_v(\phi'') + p_v(\phi'') + q_{vj}(\phi'')\} \\ \qquad \text{if } (0,0,0) \not\succ_D q_{uj}(\phi) \vee \\ \qquad (0,0,0) \not\succ_D q_{vj}(\phi) \\ C_j(\phi) \qquad \text{otherwise} \end{cases}$$

Notice that we only calculate heads and tails for the involved operations, avoiding propagating the changes to terminal nodes. This way the computation is $\mathcal{O}(1)$ and calculating the lower bound for all jobs is $\mathcal{O}(n)$.

Now, with these lower bounds for the completion times, we can approximate the TWT. When considering the makespan, the approximations are close enough to the real value to be used as a surrogate function without a significant deterioration in the quality of the final solution. However, this is not the case for the TWT and although new approximations have been proposed [15], they are too expensive to calculate and their benefits almost disappear. For this reason, we propose an improvement of the filtering mechanism introduced in [5].

This filtering mechanism consists in ordering the neighbors by their expected lower bound and evaluate them exactly in order until we find one whose estimation is bigger than the best exactly evaluated neighbor. Because we are using a lower bound, it is guaranteed that the TWT of any of the remaining neighbors cannot be better.

Our improvement compared to [5] is that instead of calculating the heads and tails for each neighbor that is exactly evaluated, we only calculate the heads and when a neighbor is finally selected, then we calculate its tails. Notice that calculating the heads is only $\mathcal{O}(t)$ whereas calculating the tails is $\mathcal{O}(tn)$ because for each operation there is a tail for each job. Also, remember that the number of neighbors was $\mathcal{O}(tm)$. Then, calculating the estimation of all neighbors is $\mathcal{O}(tmn)$ and calculating the exact values is $\mathcal{O}(t^2m)$. Thus, in the worst case (when we have to evaluate exactly all neighborhoods) the complete evaluation is $\mathcal{O}(tmn + t^2m + tn) = \mathcal{O}(t^2m)$. If instead of calculating only the heads we also calculated the tails as in [5] then the asymptotic complexity would be $\mathcal{O}(tmn + t^2mn) = \mathcal{O}(t^2mn)$. This way, our filter is much faster while it leads to the same selection. Notice that this saving can be done because to calculate the objective functions we only need the heads (which are equal to the earliest starting times). The tails are only needed to calculate the approximations in the next iteration.

IV. Experimental Study

The experimental analysis is organized as follows. First, we provide results for total weighted tardiness minimization on benchmark instances. Then, we conduct tests with the makespan as the objective function. This allows us to compare this proposal with state-of-the-art methods since, to our

TABLE I
RESULTS FOR THE TOTAL WEIGHTED TARDINESS OBJECTIVE FUNCTION

Instance	f=1.0					f=1.1					f=1.2				
	E[TWT]			Average runtime		E[TWT]			Average runtime		E[TWT]			Average runtime	
	Best	Mean	Std	Real	CPU	Best	Mean	Std	Real	CPU	Best	Mean	Std	Real	CPU
lei1	87.438	88.30	0.42	7.61	105.88	56.650	59.08	1.56	7.26	104.11	38.350	39.03	0.44	5.37	79.27
lei2	113.312	114.04	0.96	8.89	119.98	67.544	70.29	1.09	8.44	110.93	41.925	43.65	0.76	5.90	77.28
lei3	121.812	122.89	0.64	13.55	224.70	74.069	74.96	0.86	11.89	209.39	45.175	46.16	0.56	7.93	120.29
lei4	101.938	102.06	0.32	10.46	188.40	68.806	68.96	0.47	10.04	160.86	49.625	50.43	0.41	8.66	141.77
lei5	330.750	334.08	1.75	303.99	5841.40	263.450	271.03	2.89	337.51	6497.91	205.650	211.43	3.91	345.90	6848.89
lei6	338.125	344.74	3.67	413.43	8178.76	266.513	272.38	4.31	454.61	8252.17	207.350	212.58	2.86	424.66	8139.27

knowledge, the total weighted tardiness function has not been previously considered in the fuzzy setting. Finally, we report the results of a study of synergies between the components of the memetic algorithm.

All the results correspond to a parallelized implementation of the algorithm in C++ run on a Linux machine with two Intel Xeon Gold 6132 processors without Hyper-Threading (14 cores/14 threads) and 128GB RAM. The source code together with detailed results and benchmark instances can be found at https://pablogarciagomez.com/research

A. Total Weighted Tardiness

The considered instances are based on the six instances used in [16]. Instances 1 and 2 have 10 jobs, 10 machines and 40 operations; instances 3 and 4 have 10 jobs, 10 machines and 50 operations; and instances 5 and 6 have 15 jobs, 10 machines and 80 operations. In all cases there is total flexibility, i.e., any operation can be executed on any machine. To generate due dates and job weights, we have followed a standard procedure from the scheduling literature [17], so due dates are given by $d_j = f\sum_{k=0}^{n_j} \min_{\mu \in M_{jk}} E[p_{o_{jk}\mu}]$, where f is a predefined tightness factor (here $f \in \{1.0, 1.1, 1.2\}$), and the first 20% of jobs are given weight 4, the next 60% are given weight 2 and the last 20% are given weight 1.

After a preliminary experimental study, the parameters of the algorithm are set as follows: population size, $n\frac{t}{m}$; stopping criterion for the memetic algorithm, $n + m$ iterations without improvement or if the average quality of the population is equal to the best solution; stopping criterion for the tabu search, t iterations without improvement; tabu list size between t and $2t$.

Table I reports the best expected TWT as well as the mean and the standard deviation across 30 runs of the algorithm, together with the real execution time and the CPU execution time per run in seconds.

We can see that standard deviation values are really small relative to the mean. This indicates that the results are very similar between executions which is a very desirable property in stochastic search methods. As regards the execution time, the quotient between the CPU time and the real time gives us an idea about the high speedup achieved with the parallelization. Notice that a comparison with a single-threaded implementation would not have been as fair, due to the stochastic processes involved.

TABLE II
RESULTS FOR THE MAKESPAN OBJECTIVE FUNCTION

Instance	Expected makespan				Average runtime	
	BKS	Best	Mean	Std	Real	CPU
lei1	28.50	**28.50**	28.50	0.00	2.25	45.68
lei2	45.25	**45.25**	45.25	0.00	1.30	26.66
lei3	43.50	**43.50**	43.76	0.21	5.32	121.13
lei4	34.25	**34.25**	34.46	0.24	5.67	105.26
lei5	51.00	***50.75**	51.38	0.36	110.19	2344.84
lei6	50.25	***50.00**	51.22	0.54	113.18	2347.63

B. Makespan

For the sake of comparison with the the state-of-the-art methods that only consider makespan, Table II reports the results of our algorithm using this objective function and the same configuration as with the TWT. Results in bold denote that the algorithm has reached the best-known solution (BKS) and the symbol "*" denotes that it has found a better one. The algorithm reached the expected makespan of the best-known solution in the first four instances and in the fifth and sixth instances it even made a little improvement. As before, the algorithm is very stable.

It is also interesting to compare the execution times with those in Table I, which are larger. This shows that the problem with TWT is more complex and requires more computational power to be solved.

C. Synergies Study

In this section, we experimentally check if the evolutionary component and the local search component of the memetic algorithm work well together in order to find better solutions than the ones they would find separately.

We have given each component roughly the same running time as the complete memetic algorithm. To do this, in the evolutionary component we have just increased the population size and in the tabu search we have increased the number of restarts. We think this is the fairest way of doing it because we keep the stopping criteria of the original memetic algorithm.

As for the instances, we have considered the same set as in Section IV-A but only for $f = 1.0$. The obtained results can be seen in Table III. Notice that the memetic algorithm results are the same as in Table I. We use them as a reference,

TABLE III
RESULTS FOR THE SINERGIES STUDY

Instance	Memetic algorithm			Evolutionary algorithm					Tabu search				
	E[TWT]			E[TWT]			Difference (%)		E[TWT]			Difference (%)	
	Best	Mean	Std	Best	Mean	Std	Best	Mean	Best	Mean	Std	Best	Mean
lei1	87.438	88.30	0.42	88.000	89.18	1.27	0.64	1.00	95.250	103.07	3.31	8.93	16.73
lei2	113.312	114.04	0.96	113.312	115.51	3.50	0.00	1.29	129.562	137.34	2.97	14.34	20.43
lei3	121.812	122.89	0.64	121.812	123.74	4.67	0.00	0.69	138.812	153.74	4.67	13.96	25.10
lei4	101.938	102.06	0.32	105.438	107.57	1.24	3.43	5.39	108.438	118.75	3.66	6.38	16.35
lei5	330.750	334.08	1.75	342.500	371.01	31.37	3.55	11.06	498.625	533.15	15.39	50.76	59.59
lei6	338.125	344.74	3.67	345.875	370.63	35.81	2.29	7.51	479.500	518.90	12.83	41.81	50.52

calculating the relative difference between these results and the obtained by each component separately.

We can see that neither the evolutionary algorithm nor the tabu search equals the results obtained by the memetic algorithm. We can see that in general the bigger the instances, the larger the differences between the algorithms. It is also clear that the evolutionary algorithm works better than the tabu search. The main reason is that local search algorithms are very dependent on the initial solutions used to start the search whereas evolutionary algorithms explore more areas of the search space.

V. CONCLUSIONS

In this work, we have considered the fuzzy flexible job shop scheduling problem with the total weighted tardiness as the objective function to minimize. As a first contribution, we have generalized the categories of schedules existing in the traditional job shop to the case of fuzziness and flexibility. This has allowed us to propose a memetic algorithm that acts in a reduced search space. The algorithm, combining an evolutionary algorithm and a tabu search, takes advantage of their synergies to find an equilibrium between exploration and exploitation. As novel elements, we have generalized the GOX crossover operator to the case of flexibility and we have extended the concept of critically for the TWT in a fuzzy setting, an essential concept to define neighborhood functions. Also, we have proposed an improvement to a recently published neighbor filtering mechanism reducing its asymptotic complexity while keeping the same selection, thus obtaining a substantial increase in performance. Moreover, we have parallelized the algorithm to take advantage of modern CPUs and improve its scalability.

Although direct comparisons with other methods have not been possible because this particular variant of the problem has not been considered before, we have made an indirect comparison using the more traditional makespan objective. The obtained results show that we have been able to achieve state-of-the-art performance with very low variance, proving not only the search power but also the stability of the algorithm. We have even found new best solutions for two of the instances. In the case of the total weighted tardiness, this paper provides a first reference for future work in this area.

REFERENCES

[1] M. L. Pinedo, *Scheduling: Theory, Algorithms, and Systems*, 5th ed. Springer International Publishing, 2016.

[2] J. Lenstra, A. R. Kan, and P. Brucker, "Complexity of machine scheduling problems," in *Studies in Integer Programming*. Elsevier, 1977, pp. 343–362.

[3] J. J. Palacios, M. A. González, C. R. Vela, I. González-Rodríguez, and J. Puente, "Genetic tabu search for the fuzzy flexible job shop problem," *Computers & Operations Research*, vol. 54, pp. 74–89, 2015.

[4] K. Gao, Z. Cao, L. Zhang, Z. Chen, Y. Han, and Q. Pan, "A review on swarm intelligence and evolutionary algorithms for solving flexible job shop scheduling problems," *IEEE/CAA Journal of Automatica Sinica*, vol. 6, no. 4, pp. 904–916, 2019.

[5] C. R. Vela, S. Afsar, J. J. Palacios, I. González-Rodríguez, and J. Puente, "Evolutionary tabu search for flexible due-date satisfaction in fuzzy job shop scheduling," *Computers & Operations Research*, vol. 119, p. 104931, 2020.

[6] D. Dubois and H. Prade, "Fuzzy numbers: An overview," in *Readings in Fuzzy Sets for Intelligent Systems*. Elsevier, 1993, pp. 112–148.

[7] L. Zadeh, "Fuzzy sets," *Information and Control*, vol. 8, no. 3, pp. 338–353, 1965.

[8] B. Giffler and G. L. Thompson, "Algorithms for solving production-scheduling problems," *Operations Research*, vol. 8, no. 4, pp. 487–503, 1960.

[9] C. Bierwirth, "A generalized permutation approach to job shop scheduling with genetic algorithms," *OR Spektrum*, vol. 17, no. 2-3, pp. 87–92, 1995.

[10] F. Glover, "Future paths for integer programming and links to artificial intelligence," *Computers & Operations Research*, vol. 13, no. 5, pp. 533–549, 1986.

[11] M. Dell'Amico and M. Trubian, "Applying tabu search to the job-shop scheduling problem," *Annals of Operations Research*, vol. 41, no. 3, pp. 231–252, 1993.

[12] M. A. González, C. R. Vela, and R. Varela, "An efficient memetic algorithm for the flexible job shop with setup times," in *Proceedings of the Twenty-Third International Conference on Automated Planning and Scheduling, ICAPS*. AAAI, 2013.

[13] E. Nowicki and C. Smutnicki, "A fast taboo search algorithm for the job shop problem," *Management Science*, vol. 42, no. 6, pp. 797–813, 1996.

[14] É. D. Taillard, "Parallel taboo search techniques for the job shop scheduling problem," *ORSA Journal on Computing*, vol. 6, no. 2, pp. 108–117, 1994.

[15] Y. Mati, S. Dauzère-Pérès, and C. Lahlou, "A general approach for optimizing regular criteria in the job-shop scheduling problem," *European Journal of Operational Research*, vol. 212, no. 1, pp. 33–42, 2011.

[16] D. Lei and X. Guo, "Swarm-based neighbourhood search algorithm for fuzzy flexible job shop scheduling," *International Journal of Production Research*, vol. 50, no. 6, pp. 1639–1649, 2012.

[17] S. Eilon and I. G. Chowdhury, "Due dates in job shop scheduling," *International Journal of Production Research*, vol. 14, no. 2, pp. 223–237, 1976.

Estimación de las tensiones residuales a nivel de grano en una muestra cilíndrica templada de aleación de aluminio AA5083 mediante programación genética

Laura Millán
CENIM, CSIC
Madrid, España
mglauri@cenim.csic.es

Gabriel Kronberger
University of Applied Sciences Upper Austria
Hagenberg, Austria
gabriel.kronberger@fh-hagenberg.at

Ricardo Fernández
CENIM, CSIC
Madrid, España
ric@cenim.csic.es

Oscar Garnica
UCM
Madrid, España
ogarnica@ucm.es

Patricie Halodova
Centrum výzkumu Řež
Řež, República Checa
patricie.halodova@cvrez.cz

Gaspar González-Doncel
CENIM, CSIC
Madrid, España
ggd@cenim.csic.es

J.Ignacio Hidalgo
UCM
Madrid, España
hidalgo@dacya.ucm.es

***Abstract*—Las tensiones residuales se originan durante los procesos de fabricación de las piezas metálicas, por lo que su estudio es importante para evitar accidentes catastróficos durante su comportamiento en servicio. Existen dos tipos principales de tensiones residuales, según la escala de medida: macroscópicas y microscópicas. Mientras que la cuantificación de las tensiones macroscópicas es un proceso casi rutinario, la determinación de las tensiones microscópicas de los granos individuales sigue siendo una tarea pendiente. En este trabajo se presenta una aproximación que utiliza la programación genética para obtener las microtensiones residuales en los granos de una muestra cilíndrica de la aleación de aluminio AA5083. Para obtener la tensión de cada grano se estiman los valores de las microtensiones para cada orientación cristalográfica utilizando información de experimentos de difracción de neutrones y de electrones retrodispersados. Esta información incluye mapas de orientación de una sección normal a los ejes del cilindro (orientaciones individuales) y las características de cada grano. Con todo ello, suponemos que las microtensiones residuales de cada grano pueden expresarse como una función basada en estas variables y utilizamos la programación genética para encontrar esta expresión.**

***Index Terms*—Programación Genética, difracción de neutrones, EBSD, orientación, distancia interplanar, microtensiones**

I. Introducción

Las tensiones residuales son aquellas que aparecen en los materiales como consecuencia de los procesos de mecanizado. Se generan cuando se produce un cambio dimensional no homogéneo (deformaciones y/o dilataciones), y permanecen en el material una vez finaliza el procesado. Las tensiones residuales en los materiales se diferencian según la escala dimensional en la que se manifiestan. Las tensiones macroscópicas (M-RS), se pueden determinar sin gran complejidad. Sin embargo, existen otras tensiones, las microscópicas, cuya magnitud aún se desconoce, pero su influencia en problemas como la rotura de temprana de componentes metálicos debido a grietas generadas por estos campos de tensión también puede ser relevante. Para entender el origen de las tensiones microscópicas, es necesario profundizar en la microestructura de los materiales. Estas tensiones microscópicas se clasifican en intergranulares (tipo II) e intragranulares (tipo III). Las tensiones microscópicas de tipo II (o m-RS), que son las que interesan en esta investigación, varían entre granos vecinos como consecuencia del diferente grado de deformación plástica (y elástica) que ha sufrido un grano respecto a los demás de su entorno durante un determinado proceso metalúrgico (por ejemplo, la laminación). Este estudio se centra en la determinación de estas m-RS. El objetivo de es desarrollar una metodología para el mapeo de las m-RS, particularmente en componentes de aluminio. El procedimiento aplica un enfoque de programación genética (GP) a los datos de difracción de neutrones adquiridos en grandes instalaciones europeas. Nuestra propuesta se basa en el análisis de los desplazamientos y ensanchamientos de los picos de difracción y la correspondiente microestructura del material, determinada por EBSD (Difracción de Electrones Retrodispersados). Se aplica la GP para buscar una expresión que relacione el patrón de difracción medido con los parámetros microestructurales de los granos, prediciendo el desplazamiento del pico en función de estas variables microestructurales.

El resto del artículo se organiza como sigue. En la sección II se analizan trabajos anteriores relacionados con el tema. En la sección III se explican los métodos utilizados para recoger los datos y se describen los detalles de la aplicación de la GP. La sección IV contiene los resultados y, por último, la sección V resume el resultado principal y describe el trabajo en curso.

II. Estado del arte

En cuanto al uso de la computación evolutiva para resolver problemas de materiales, los algoritmos genéticos se han empleado para optimizar otros procesos termomecánicos como la soldadura por fricción (FSW) en muestras de AA2014 y AA6061[1]. Además, las redes neuronales entrenadas mediante algoritmos evolutivos pueden modelar las propiedades mecánicas de los materiales [2]. Nuestro grupo ha demostrado la presencia de m-RS en muestras soldadas [3]. En particular, pudimos reproducir el perfil M-RS a través de las soldaduras utilizando algoritmos evolutivos [4]. Sin embargo, aún queda mucho camino por recorrer para desarrollar una metodología completa para calcularlos. El promedio de m-RS de la aleación de aluminio AA2014 se estimó en un trabajo más reciente [5] en el que se utilizó una aleación endurecible por envejecimiento. Sin embargo, aún no se ha desarrollado una metodología para mapear el m-RS de todos los granos que forman el policristal. Por otro lado, los tratamientos termomecánicos pueden generar precipitados de distinta naturaleza en función de los elementos de aleación. Estos precipitados son agregados que aparecen en la microestructura y tienen una composición diferente a la de los granos de aluminio iniciales. Su presencia modifica la microestructura, distorsiona la red cristalina y puede alterar las propiedades del material. Así, las m-RS de los granos de aluminio originales puede verse alteradas por la presencia de estos precipitados. No obstante, la estimación de un valor medio de las m-RS fue una contribución importante de [5]. Recientemente, nuestro grupo mejoró el método de equilibrio de tensiones para determinar la M-RS [6] considerando las diferencias en las microestructuras internas y externas, y el estado de tensiones triaxial. Además, resolvimos el problema de los precipitados utilizando una aleación no endurecible, obteniéndose valores más robustos. En este trabajo, proponemos continuar esta línea y combinar todos los métodos mencionados anteriormente utilizando GP para obtener un valor de m-RS de cada uno de estos granos. Siendo ésta la primera aproximación conocida que se propone para identificar las m-RS a nivel de grano.

III. Metodología

Para este estudio se eligió una muestra cilíndrica de aleación de aluminio 5083. Se trata de una aleación comercial policristalina y monofásica; es decir, tiene una microestructura formada por un gran número de granos de composición similar y distintas orientaciones cristalográficas. Cada uno se caracteriza por su tamaño y orientación, y suelen están rodeados por un grupo de granos vecinos que pueden tener la misma o diferente orientación. Existen cuatro orientaciones cristalográficas principales descritas por los índices de Miller (hkl)[7]: 111, 200, 220 y 311.

El método de difracción de electrones retrodispersados (EBSD) permite determinar las orientaciones individuales de los granos. Esta técnica proporciona un mapa de granos como el representado en la Figura 1. En ella se muestra la microestructura de la cara axial de un cubo extraído a 12mm del centro de la muestra cilíndrica, que se sometió a un

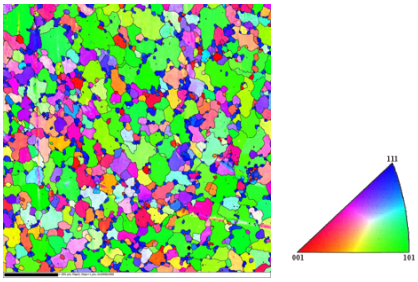


Fig. 1: Mapa de las orientaciones de los granos individuales obtenido por EBSD a 12 milímetros del centro de la muestra. La normal a la superficie observada es paralela a los ejes del cilindro

tratamiento termomecánico de extrusión y temple. Es importante destacar que los granos con distinta orientación presentan un comportamiento mecánico diferente bajo la acción de una fuerza uniaxial externa (anisotropía cristalina). Esto provoca la generación de m-RS cuando el material se somete a un proceso termomecánico que desarrola una deformación plástica (no homogénea) en el mismo. Se deduce de ello que debe existir un patrón que rige la colocación de los granos en función del tratamiento y la orientación de los mismos, siendo la microestructura resultante la que describe estas m-RS.

Además de lo que se ha comentado en referencia al mapa de orientaciones de la Figura 1, las mediciones de EBSD también proporcionan una lista completa de los granos y sus características: número de granos, áreas y diámetros de cada uno de ellos, número de granos vecinos, relación de aspecto, pendiente y orientación cristalográfica. Estos parámetros están presentes para cada grano y tienen asociado un valor de m-RS que depende de su comportamiento mecánico. Por tanto, debe haber una función que los relacione y se cumpla para todos los granos de la misma orientación.

Por otro lado, la difracción de neutrones proporciona información sobre la tensión residual global de los materiales. El estado de m-RS de un gran colectivo de granos con orientación cristalográfica común se modifica con la posición y la anchura del pico. Hay cuatro picos de alta intensidad: 111, 200, 220 y 311, que corresponden a las orientaciones preferidas (textura) del material. El espaciado de la red o distancia interplanar, $d_{x_i}^{hkl}$, se utiliza para calcular la tensión. Este espaciado de red es la distancia entre dos planos cristalinos paralelos de átomos que tienen la misma orientación. Se asume aquí que si el m-RS de los granos individuales depende de la microestructura, debe haber una función matemática que relacione estas tensiones con los parámetros microestructurales. Teniendo en cuenta el comportamiento mecánico de los granos, esta función debería ser la misma para granos con la misma orientación. Formulamos el objetivo de encontrar esta expresión matemática como un problema de regresión simbólica y utilizamos la GP para resolverlo. Las variables independientes para el modelo son los parámetros microestructurales proporcionados por el EBSD. La variable dependiente es el valor del espaciado

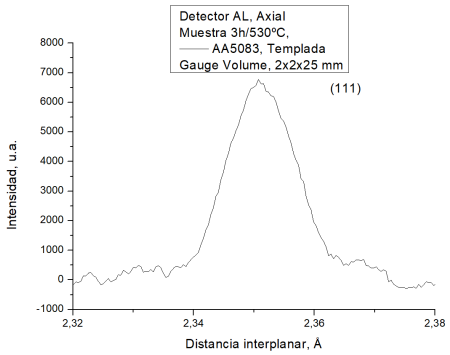


Fig. 2: Distribución de intensidades del pico 111 extraído del espectro a 12mm del centro

de la red, $d_{x_i}^{hkl}$, para cada grano como sustituto del m-RS. Sin embargo, es imposible medir m-RS para los granos individuales sin un error asociado enorme, sino sólo el pico de difracción (medido) proporcionado por un gran número de granos con un error mucho menor. En su lugar, calculamos una distribución de difracción predicha a partir de los valores $d_{x_i}^{hkl}$, individuales predichos e intentamos minimizar la diferencia entre la distribución medida mediante difracción de neutrones y la predicha. Así, el modelo de regresión simbólica obtenido mediante GP nos permite estimar el estado m-RS para cada grano a partir de los parámetros microestructurales.

La difracción de neutrones permite determinar con alta precisión el espaciado de red, $d_{x_i}^{hkl}$, de diferentes planos (hkl) que es el requisito para calcular el nivel de deformación (y tensión) desarrollada entre los granos. Debido a su alta capacidad de penetración se pueden realizar mediciones en el interior de materiales, para lo cual no se requiere una preparación superficial del material.

El instrumento utilizado proporciona las medidas en unidades TOF (tiempos de vuelo de los neutrones). La conversión a unidades de distancia interplanar se realizó utilizando los parámetros del instrumento, los detectores y la expresión:

$$d_{x_i}^{hkl} = \frac{(N + D_0) \cdot \tau}{C \cdot L \cdot sin(\theta))} \quad (1)$$

where

- $d_{x_i}^{hkl}$ es la distancia interplanar para cada componente hkl en la posición i y en el eje x.
- C es una constante cuyo valor es 505556
- D_0 es el tiempo de retardo inicial
- τ es la anchura del canal
- L es la distancia total del vuelo de los neutrones desde la fuente al detector
- θ es el ángulo de Bragg
- N es el número de canales por los que pasan los neutrones

Como se mencionó, la implementación de GP tiene como objetivo relacionar cada grano de la lista obtenida por EBSD, con un valor de la distancia interplanar, $d_{x_i}^{hkl}$, obtenida por difracción de neutrones. Por lo tanto, se busca obtener distribuciones de tensión (similares a los perfiles de los picos experimentales) que se puedan describir mediante una expresión cuyos términos sean los parámetros microestructurales proporcionados por la técnica EBSD. Para simplificar el problema, se asumen las siguientes aproximaciones:

- Para un primer análisis, solo se seleccionó el pico de orientación 111, debido a su particular comportamiento mecánico (son los granos de mayor dureza) en la dirección axial. Por tanto, se asume que la distribución de tensiones la producen 626 granos en este caso.
- La simulación estudiará la cara axial del cubo nº 3, es decir, la posición a 12mm del centro del cilindro inicial.
- Todos los granos 111 en la lista de EBSD deben contribuir proporcionalmente al perfil observado en la Figura 2. Esto implica que la ecuación de la distancia interplanar predicha debe ser la misma a pesar del grano (es decir, se asume que los granos tienen aproximadamente la misma participación en el pico de difracción).
- Se considera que el ruido o las interferencias en el patrón de difracción que proceden del instrumento de medición no tendrán influencia en los resultados. Y, por ello, no será tenido en cuenta en el modelo.

De esta forma, la fórmula encontrada usando el modelo de regresión simbólica nos permite calcular la distancia interplanar, $d_{x_i}^{hkl}$, a partir de los parámetros descritos anteriormente. Estos valores se utilizarán para obtener la tensión total, $\sigma_{x_i}^{T,hkl}$, para cada grano. El proceso para calcular estas tensiones totales se describe en los siguientes pasos. Primero, se utilizó el método de equilibrio de tensiones explicado en [6], para obtener la tensión macroscópica, M-RS, y el valor de la distancia interplanar correspondiente al estado libre de tensiones, d_0^{hkl}, para todas las orientaciones. Los resultados fueron ya discutidos con anteriorodad en [6]. Luego, se obtiene la deformación o elongación, $\epsilon_{x_i}^{hkl}$, de cada grano, a partir de los valores del estado libre de tensiones y la distancia interplanar predicho:

$$\epsilon_{x_i}^{hkl} = \frac{d_{x_i}^{hkl} - d_0^{hkl}}{d_0^{hkl}} \quad (2)$$

dónde

- $\epsilon_{x_i}^{hkl}$ es la deformación que sufren los granos con orientación hkl en el eje x.
- $d_{x_i}^{hkl}$ es la distancia interplanar experimental de los granos con orientación hkl en el eje x.
- d_0^{hkl} es el valor de la distancia interplanar correspondoente al estado libre de tensiones de los granos con orientación hkl.

Sustituyendo estos valores de deformación en la siguiente ecuación, se obtienen las tensiones totales $\sigma_{x_i}^{T,hkl}$ en la posición i:

$$\sigma_{x_i}^{T,hkl} = E^{hkl} \epsilon_{x_i}^{hkl} \quad (3)$$

dónde

- $\sigma_{x_i}^{T,hkl}$ es la tensión total para la posición i y el componente de textura hkl (orientación) en el eje x

Parametro	Valor
Set de funciones	+, -, *, log(x), exp(x)
Set de variables:	area, número de vecinos, relación de aspecto, inclinación de los granos, constantes aleatorias
Creación del árbol	PTC2
Tamaño. población	500
Max. generaciones	1000
Max. tamaño árbol	50 nodos
Max. altura árbol	12 niveles
Probabilidad de mutación	15%
Probabilidad de cruce	100%
Selección	masculina: fitness-proporcional femenina: aleatorio
Selección de descendientes	estricta
Función Fitness	Diferencias absolutas entre distribuciones de intensidad (ver más abajo)

TABLE I: Parámetros usados en la GP

- E^{hkl} es el módulo elástico y es constante para cada orientación hkl.

Finalmente, la M-RS calculada previamente se utilizó para obtener la m-RS, considerando que la tensión total es la suma de las tensiones macro y microscópicas.

$$\sigma_{x_i}^{T,hkl} = \sigma_{x_i}^{M} + \sigma_{x_i}^{m,hkl} \tag{4}$$

where

- $\sigma_{x_i}^{m,hkl}$ es la micro-tensión para la posición i y el componente de textura hkl en la dirección x.
- $\sigma_{x_i}^{M}$ es la Macro-tensión para la posición i en la dirección x.

Programación Genética

Se usa GP con el creador de árbol probabilístico (PTC2) [8] para inicializar la población y la selección específica de género, donde el primer padre se selecciona con una probabilidad proporcional a su aptitud y el segundo padre se selecciona aleatoriamente [9]. Todos los hijos se producen a través del cruce de subárboles y una mutación aleatoria opcional de la función o de símbolos de la misma. Solo los hijos con una aptitud física igual o mejor que ambos padres se agregan a la próxima generación (selección de descendientes) [10]. La selección estricta de la descendencia, en la que los hijos deben ser mejores que ambos padres, garantiza una presión de selección suficiente incluso con una presión bastante baja en la selección de los padres [10]. El mejor individuo se mantiene como en la élite.

La tabla I muestra todos los parámetros de GP que se usaron. En experimentos preliminares se observó que valores específicos de los parámetros utilizados no tienen un gran impacto en los resultados. Se probaron varios tamaños de población diferentes, así como límites de tamaño de árboles. Al final, se eligieron los valores de los parámetros que sabemos que funcionan de forma robusta para muchas instancias de problemas diferentes y un tamaño de árbol máximo que garantiza que el modelo resultante aún sea legible. En los ensayos, se visualizó la curva que se muestra en la Figura 4 para el mejor modelo hasta ahora en la ejecución de GP. Constantemente se observa que GP rápidamente (en unos pocos minutos) converge a una curva que esta cerca de la curva objetivo. Las mediciones son ruidosas y el ajuste del modelo GP fue con frecuencia lo suficientemente bueno incluso después de unos minutos de tiempo de ejecución. Para los experimentos, se aumentó el número de generaciones para dar como resultado un tiempo de ejecución de aproximadamente una hora.No se ha intentado optimizar estos valores de parámetros, ya que el problema se puede resolver de forma coherente utilizando los parámetros elegidos y este no es un estudio sobre la optimización de la eficiencia de GP para este tipo de problema.

Algorithm 1: Algoritmo para calcular la distribución de intensidades estimada a partir de un modelo para predecir los valores de la distancia interplanar.

Input: Modelo de regresión simbólica m(), dataset con los granos y sus características medidos por EBSD, dataset $\mathcal{D}$ con la actual distancia interplanar medida por difracción de neutrones

Output: Estimación del número de granos para cada valor de la distancia interplanar $d_{x_i}^{khl}$

deviation $\leftarrow 10^{-4}$

scaling_factor $\leftarrow \frac{1}{\text{deviation}\cdot\sqrt{2\cdot\pi}}$

foreach *grain g = (area, numNeighbours, aspectRatio, slope)* **do**

  $d_{x_i}^{pred} \leftarrow m(g)$ `/* Predicted lattice spacing value */`

  foreach $d_{x_i} \in \mathcal{D}$ **do**

    intensity$[d_{x_i}]+=$ scaling_factor $\cdot \frac{\exp(-0.5\cdot(d_{x_i}-d_{x_i}^{pred})^2}{\text{deviation}^2}$

  end

end

La función de fitness es específica para esta aplicación, ya que usamos GP para predecir una cantidad que no es directamente medible, la distancia interplanar de los granos. En cambio, necesitamos estimar una distribución de intensidad para la distancia interplanar predicha y hacer coincidir esta distribución con la distribución medida por difracción de neutrones. Suponemos que la distribución medida es una suma de picos individuales, uno para cada grano, cada uno de los cuales puede aproximarse mediante curvas de campana gaussianas con pequeñas variaciones. En consecuencia, calculamos el espectro estimado a partir de la distancia interplanar predicha a través de una convolución. El pseudocódigo en el algoritmo 1 muestra el procedimiento de convolución para un modelo de regresión simbólico dado $m()$ para la distribución discretizada. En las ejecuciones, se experimentó con diferentes valores de varianza y se encontró que 10^{-4} funcionaba bien. Después de la convolución, se calculó la suma de las diferencias absolutas de intensidad para la distribución estimada y medida. La suma negativa de las diferencias absolutas se utiliza como función de

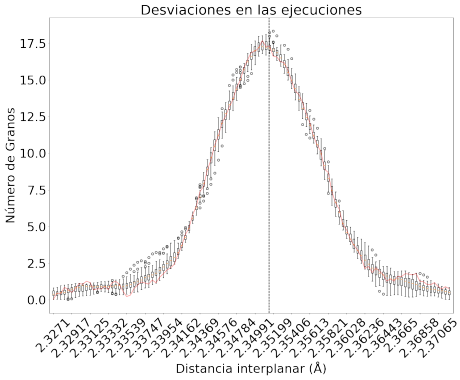


Fig. 3: Desviaciones en las ejecuciones de GP e intensidades medidas.

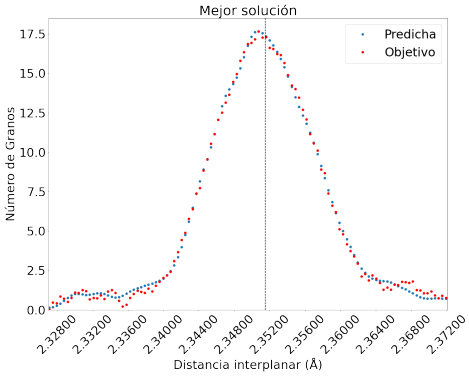


Fig. 4: Mejor solución encontrada por GP para la distribución de granos 111 a 12 mm del centro

actitud a maximizar. Para la implementación y los experimentos, se usó HeuristicLab [1] un entorno de código abierto para la optimización heurística, que incluye una implementación de GP y regresión simbólica. La función de fitness personalizada se implementa mediante la funcionalidad de secuencias de comandos de HeuristicLab.

IV. Resultados

Se realizaron treinta ejecuciones de GP en un procesador Intel Core i5 con 8 GB de RAM, con sistema operativo Windows 10. En promedio, se evaluaron aproximadamente 920000 soluciones que requerían 40 minutos. Los resultados se muestran en la Figura 3. Cada solución da una distribución de los granos que se ajusta al perfil de difracción. La figura resume los resultados de las 30 distribuciones. La línea roja es el perfil real del pico 111 axial, que es la distribución objetivo de este documento. En la Figura 3, el eje x indica el valor de la distancia interplanar $d_{x_i}^{hkl}$, correspondiente al rango de difracción del haz de neutrones con los granos 111. El eje y representa el número de granos 111 axiales asignados por GP a cada valor de la distancia interplanar, $d_{x_i}^{hkl}$. Cada diagrama de cajas agrupa los valores correspondientes a cada una de las treinta ejecuciones. Una línea dentro de cada caja indica el promedio de estos treinta valores y la barra su desviación estándar. Se analizaron los valores de error con respecto a la distribución objetivo, y se seleccionó la distribución con las diferencias menores con respecto al perfil real, mostrándose en la Figura 4.

La expresión correspondiente a la mejor solución aparece simplificada en la siguiente ecuación [2]:

$$\begin{aligned} d_{x_i}^{hkl} &= -1.18440 \cdot 10^{-7} I_n AR_n S_n + \\ &+ 6.99501 \cdot 10^{-8} AR_n + 8.15099 \cdot 10^{-6} S_n - \\ &- 4.95588331133054 \cdot 10^{-7} (0.68927 A_n + \\ &+ 31.48632 AR_n^2 + 16.60289)(0.57222 A_n - \\ &- 18.74758 I_n - 2.39264 AR_n - 0.96370 S_n - \\ &- 25.90657) + 2.3403 \end{aligned} \quad (5)$$

[1] https://dev.heuristiclab.com

[2] La salida debe escalarse linealmente al rango de la medición de difracción

donde,

- A_n es el area para cada grano n
- I_n es el número de vecinos para cada grano n
- AR_n es la relación de aspecto de cada grano n
- S_n es la inclinación con respecto a la medida de cada grano n

Además, el parámetro microestructural que aparece con mayor frecuencia es la relación de aspecto. Este es un hecho razonable ya que la compresión o tracción de un grano, y con ello su forma, depende de la tensión a la que está vinculado. Además, recordando el mapa de orientaciones de la figura 1 vemos que efectivamente los granos 111 axiales poseen tamaños en su mayoría pequeños pero con una forma e inclinación muy dispar, es decir, presentan mucha variación en su relación de aspecto. Incluso, podemos encontrar granos 111 grandes y aislados con formas distintas. Esto indica que el método coincide con la realidad y es robusto. Por lo que la expresión obtenida se puede aplicar para calcular las m-RS utilizando los parámetros microestructurales de EBSD.

Los valores de los parámetros microestructurales para los granos 111 fueron reemplazados en la expresión obtenida, y su m-RS se calculó usando las ecuaciones descritas en la Sección III. Algunos de los valores obtenidos se muestran en la Figura 5. Para saber si esta solución es físicamente factible, podemos analizar si las tensiones asignadas a cada grano son coherentes con otras variables.

En este sentido, se sabe que los Módulos de Elasticidad E^{xyz}, para cada una de las cuatro familias de granos son: E^{111} = 75.5GPa, E^{200} =63.2 GPa, E^{220} = 72.0 GPa y E^{311} = 69.0 GPa. Esos valores nos informan sobre la rigidez de los granos, siendo los granos de 111 los más rígidos y los 200 los menos rígidos. Teniendo en cuenta estas consideraciones, podemos ver en la Figura 5 que los granos 111 rodeados por sí mismos tienen más tensión como ocurre con el granos de ID.836. Sin embargo, si los granos vecinos son menos rígidos, como los granos de orientación 200 o 311, por ejemplo, vecinos de los granos de ID.99 y 781, la tensión es menor. No obstante, en algunos casos, el comportamiento es diferente. Este es, por ejemplo, el caso del

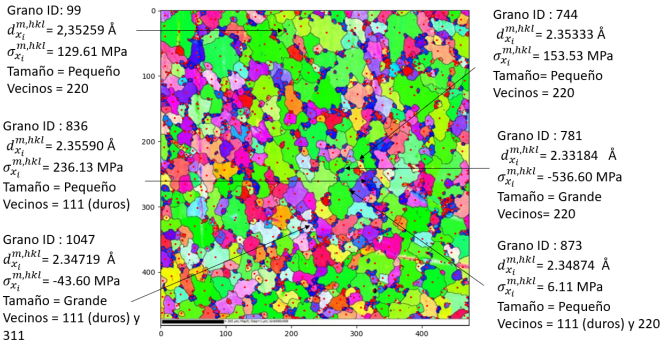


Fig. 5: Selección de los granos 111 con distancia interplanar obtenido con GP y su m-RS

grano de ID.1047 (en la parte inferior izquierda) de la Figura 5. Esto parece deberse a su tamaño; los granos más grandes presentan menos tensión que los granos más pequeños. Por tanto, en algunos granos se puede observar la influencia de ambos factores. Este hecho parece coincidir con la expresión $d_{x_i}^{hkl}$ obtenida por GP, donde la relación de aspecto y el número de vecinos son los términos más frecuentes. En consecuencia, este modelo determina los valores de m-RS de acuerdo con lo que teóricamente podría esperarse.

V. Conclusiones y trabajo en progreso

Este artículo propone un método para estimar las micro tensiones residuales en una muestra cilíndrica templada de aleación de aluminio AA5083 usando GP. Esta es la primera vez que las m-RS se estiman numéricamente a nivel de grano. Esto no se puede resolver directamente utilizando un enfoque de aprendizaje supervisado como los habituales y requiere una función de aptitud personalizada de GP para vincular los parámetros microestructurales de los granos al perfil del pico adquirido mediante difracción de neutrones.

El bajo valor de error entre el perfil de pico previsto y el objetivo demuestra que el método es válido y de calidad. Esta función estimada generado valores de $d_{x_i}^{hkl}$ dentro del rango esperado. No obstante, la expresión identificada solo es válida para una dirección espacial dentro de la red cristalográfica. En consecuencia, como paso siguiente, se utilizará la combinación de los perfiles de los picos de todas las orientaciones (111, 200, 220 y 311), así como los datos microestructurales correspondientes a otras direcciones (axial, radial y tangencial) para identificar de forma más general las m-RS de los granos. Suponemos que esta misma aproximación por GP puede aplicarse nuevamente y de forma general para este cometido. Los resultados de las m-RS se compararán con las características microestructurales en un estudio posterior.

Aparentemente, GP indica que el tamaño y tipo de granos vecinos y, así como, la relación de aspecto son los factores más importantes que influyen en los valores de m-RS en granos de 111 en esta posición. En trabajos futuros se deberá incluir una visión completa de todas las orientaciones y posiciones a lo largo del diámetro de las piezas.

Agradecimientos

Este trabajo ha sido apoyado por las ayudas FEDER de la Comunidad de Madrid Y2018 / NMT-4668 (Micro-Stress-MAP-CM) y MAT2017-83825-C4-1-R. También se agradece al FLNR-JINP por el tiempo de haz asignado en el instrumento FSD y al Centrum Výzkumu Řež, en Praga, por el mapa EBSD y el análisis microestructural.

References

[1] D Vijayan and P Abhishek. Multi objective process parameters optimization of friction stir welding using nsga–ii. In *IOP Conference Series: Materials Science and Engineering*, volume 390, page 012087, 2018.

[2] S. K. Karak, S. Chatterjee, and S. Bandopadhyay. Mathematical modelling of the physical and mechanical properties of nano-Y2O3 dispersed ferritic alloys using evolutionary algorithm-based neural network. *Powder Techn.*, 274:217–226, 2015.

[3] F Cioffi, JI Hidalgo, Ricardo Fernández, T Pirling, B Fernández, D Gesto, I Puente Orench, Pilar Rey, and Gaspar González-Doncel. Analysis of the unstressed lattice spacing, d0, for the determination of the residual stress in a friction stir welded plate of an age-hardenable aluminum alloy–use of equilibrium conditions and a genetic algorithm. *Acta Materialia*, 74:189–199, 2014.

[4] J Ignacio Hidalgo, Ricardo Fernández, J Manuel Colmenar, Florencia Cioffi, José L Risco-Martín, and Gaspar González-Doncel. Using evolutionary algorithms to determine the residual stress profile across welds of age-hardenable aluminum alloys. *Applied Soft Computing*, 40:429–438, 2016.

[5] R. Fernández, S. Ferreira-Barragáns, J. Ibáñez, and G. González-Doncel. A multi-scale analysis of the residual stresses developed in a single-phase alloy cylinder after quenching. *Materials and Design*, 137:117–127, 2018.

[6] L Millán, G Bokuchava, R Fernández, I Papushkin, and G González-Doncel. Further insights on the stress equilibrium method to investigate macroscopic residual stress fields: case of aluminum alloys cylinders. *Journal of Alloys and Compounds*, page 158506, 2020.

[7] H-J Bunge. *Texture analysis in materials science: mathematical methods*. Elsevier, 2013.

[8] Sean Luke. Two fast tree-creation algorithms for genetic programming. *IEEE Transactions on Evolutionary Computation*, 4(3):274–283, 2000.

[9] Stefan Wagner and Michael Affenzeller. Sexualga: Gender-specific selection for genetic algorithms. In *Proceedings of the 9th World Multi-Conference on Systemics, Cybernetics and Informatics (WMSCI)*, volume 4, pages 76–81, 2005.

[10] Michael Affenzeller, Stephan Winkler, Stefan Wagner, and Andreas Beham. *Genetic Algorithms and Genetic Programming: Modern Concepts and Practical Applications*. Chapman & Hall/CRC, 1st edition, 2009.

Towards Automatic Bayesian Optimization: A First Step Involving Acquisition Functions*

*Note: The full contents of this paper have been published in the volume Lecture Notes in Artificial Intelligence 12882 (LNAI 12882)

Luis C. Jariego Pérez, Eduardo C. Garrido-Merchán
Universidad Autónoma de Madrid
Madrid, Spain
luis.jariego@estudiante.uam.es, eduardo.garrido@uam.es

***Abstract*—Bayesian Optimization (BO) is the state of the art technique for the optimization of black boxes, i.e., functions where we do not have access to their analytical expression nor its gradients, are expensive to evaluate and its evaluation is noisy. A BO application is automatic hyperparameter tuning of machine learning algorithms. BO methodologies have hyperparameters that need to be configured such as the surrogate model or the acquisition function (AF). Bad decisions over the configuration of these hyperparameters implies obtaining bad results. Typically, these hyperparameters are tuned by making assumptions of the objective function that we want to evaluate but there are scenarios where we do not have any prior information. In this paper, we propose an attempt of automatic BO by exploring several heuristics that automatically tune the BO AF. We illustrate the effectiveness of these heurisitcs in a set of benchmark problems and a hyperparameter tuning problem.**

DOI: https://doi.org/10.1007/978-3-030-85713-4_16

Diseñando múltiples genes para una proteína mediante el algoritmo memético SFLA

Belen Gonzalez-Sanchez
Depto. Tecn. Computadores y Comunic.
Universidad de Extremadura
Cáceres, España
belengs@unex.es

Miguel A. Vega-Rodríguez
Depto. Tecn. Computadores y Comunic.
Universidad de Extremadura
Cáceres, España
mavega@unex.es

Sergio Santander-Jiménez
Depto. Tecn. Computadores y Comunic.
Universidad de Extremadura
Cáceres, España
sesaji@unex.es

***Resumen*—Una estrategia para aumentar los niveles de expresión de las proteínas, e intentar así satisfacer uno de los objetivos de la biología sintética, es la codificación de la proteína con múltiples copias del mismo gen. Esta codificación resulta una tarea compleja y con un proceso biológico a vigilar: la recombinación homóloga. Cuando secuencias de genes idénticas están cercanas pueden inducir recombinación homóloga y tener como resultado la pérdida de parte de las secuencias. Este es un proceso que se intenta evitar codificando las múltiples copias con los diferentes codones sinónimos que representan a cada aminoácido. De esta forma, es posible codificar múltiples copias de un mismo gen teniendo como resultado secuencias diferentes. Sin embargo, el uso de un codón sinónimo u otro también influye directamente en los niveles de expresión de la proteína, estando unos mejor adaptados que otros al organismo, y es recomendable el uso de los codones sinónimos mejor adaptados (con mayor frecuencia de uso). Esta confrontación, entre codificar las múltiples secuencias eligiendo siempre los codones sinónimos mejor adaptados pero a la vez forzar a que las múltiples secuencias contengan las máximas diferencias posibles, establece la base de un problema de optimización multiobjetivo donde mejorar un objetivo implica empeorar otro y viceversa. En este trabajo desarrollamos una implementación del algoritmo memético SFLA (*Suffled Frog Leaping Algorithm*) para resolver este problema con muy buenos resultados. Además analizamos diferentes variantes de la definición del problema, viendo cuál de ellas es mejor.**

***Palabras clave*—Diseño de múltiples genes, proteína, algoritmo memético, algoritmo SFLA.**

I. Introducción

Uno de los retos actuales que plantea la biología sintética es la codificación de proteínas, y en particular, la codificación de proteínas con múltiples copias con el propósito de maximizar los niveles de expresión de dichas proteínas. La codificación con varias copias sigue una estrategia ampliamente utilizada [1], [2] en la que se espera que los niveles de expresión aumenten proporcionalmente al número de copias [3] que se introducen, aunque esto no suceda en todos los casos [4].

Un inconveniente de codificar una proteína con múltiples copias es que secuencias o subsecuencias idénticas que se encuentren muy próximas pueden inducir el proceso biológico de recombinación homóloga, que tiene como consecuencia la pérdida de genes [5]. Existen múltiples estudios que analizan cuál es la longitud mínima de estas subsecuencias idénticas que inducen la recombinación homóloga, sin embargo no se ha llegado a determinar una longitud exacta [6], [7] para todos los organismos. Aun así, donde todos coinciden es en la importancia de minimizar la longitud de subsecuencias idénticas tanto como sea posible.

Aún con estas limitaciones, es posible diseñar múltiples secuencias que codifican una proteína, CDSs (*CoDing Sequences*), con diferencias entre ellas usando los codones sinónimos que representan cada aminoácido. Sin embargo, no todos los codones sinónimos aportan por igual para maximizar los niveles de expresión de una proteína [8] y cada organismo tiene preferencia por ciertos codones sinónimos mejor adaptados, siendo estos preferiblemente los elegidos para la codificación de las secuencias. Por lo tanto, la codificación de múltiples copias con diferencias entre secuencias pero a la vez utilizando los codones mejor adaptados supone la optimización de objetivos en conflicto, donde mejorar uno implica empeorar otro y viceversa.

Para la codificación de proteínas con múltiples secuencias se define un objetivo para optimizar el uso de los codones mejor adaptados (*Codon Adaptation Index*, CAI) y dos objetivos centrados en crear diferencias entre las CDSs que forman una solución: el primero de ellos maximiza la distancia de Hamming entre CDSs, mientras que el segundo se focaliza en romper subsecuencias idénticas encontradas en una misma CDS (subsecuencias repetidas) o entre 2 CDSs (subsecuencias en común), minimizando la longitud de estas subsecuencias.

En la literatura examinada se encuentran diferentes trabajos que analizan individualmente la optimización del CAI [9]–[11] e incluso herramientas web para la codificación de secuencias optimizando el CAI, como COOL [12] o D-Taylor [13]. En relación a la optimización multiobjetivo aplicada a la codificación de múltiples copias, que es el propósito de este trabajo, se ha encontrado una publicación reciente [14] y será utilizada como referencia en las comparativas para evaluar la calidad de nuestra propuesta.

En este trabajo realizamos varias contribuciones en cuanto a la definición del problema y al algoritmo respecto a la literatura previa. En primer lugar, el uso de un algoritmo diferente que combina la búsqueda local junto a la búsqueda global como es el algoritmo SFLA (*Shuffle Frog Leaping Algorithm*), y que se ha adaptado al campo de la optimización multiobjetivo. Otra de las contribuciones es la redefinición del

problema, ya que al inicio de la fase experimental encontramos una correlación entre dos de los objetivos optimizados y planteamos el problema como la optimización de solamente dos de los objetivos, eliminando uno de los objetivos correlacionados. Por último, realizamos una comparación con el estado del arte [14], utilizando 3 métricas de calidad multiobjetivo.

El resto de la publicación queda estructurada en las siguientes secciones. La Sección II recoge la definición formal del problema. A continuación, la Sección III detalla la implementación realizada del algoritmo SFLA así como las adaptaciones llevadas a cabo para la optimización multiobjetivo. En la Sección IV se establecen los parámetros con los que se ejecutan los experimentos, se analizada detalladamente la redefinición del problema y, por último, se realiza una comparativa con el estado del arte. Para finalizar, la Sección V expone las conclusiones finales y posibles trabajos futuros.

II. Definición del problema

En nuestro problema de optimización multiobjetivo una solución es un conjunto de secuencias (CDSs) que codifican mediante codones la secuencia de aminoácidos que forman una proteína. El número de secuencias que codifica cada proteína es establecido por el usuario. En la Figura 1 se muestra un ejemplo de solución para una secuencia de 5 aminoácidos con 3 CDSs. Como se puede apreciar, la representación final de una solución es un conjunto de cadenas de caracteres, donde todas las cadenas de caracteres (CDSs) tienen igual longitud. Respecto a las funciones objetivo que evalúan las soluciones, este problema define tres: una función para optimizar el uso de los codones mejor adaptados (mCAI) y otras dos para optimizar las diferencias entre secuencias (mHD y MLRCS). Las tres funciones objetivo se explican a continuación.

Con el objetivo de que todas las CDSs que forman una solución tengan el máximo índice de adaptación de codón, la primera función objetivo (mCAI) se focaliza en la CDS_i (de un total de I CDSs) con el valor mínimo de CAI, como indica la Ecuación 1:

$$mCAI = \min_{1 \le i \le I} \quad CAI(CDS_i). \tag{1}$$

Es decir, el objetivo es maximizar el valor mínimo encontrado. Previamente, para cada una de las CDSs, se calcula el valor de CAI según la Ecuación 2:

$$CAI(CDS_i) = \sqrt[N]{\prod_{n=1}^{N} W(codon_{i,n})}, \tag{2}$$

donde $W(\cdot)$ retorna el peso de cada codón n del CDS_i, peso basado en los datos de frecuencia de uso que proporciona el trabajo relacionado [14], y N es el número de codones que forman una CDS.

La segunda función objetivo (mHD) está basada en el cálculo de la distancia de Hamming (HD) entre secuencias. Para cada par de CDSs de la solución, se calcula HD como indica la Ecuación 3:

CDS_1	GGA	GCU	UGC	UUA	GCC
CDS_2	GGU	GCC	UGU	CUU	GCU
CDS_3	GGG	GCG	UGU	CUG	GCG
Aminoácidos	G	A	C	L	A

Figura 1. Ejemplo de codificación de una secuencia de aminoácidos (proteína) con 3 CDSs y representación del cálculo de la función MLRCS: resaltadas en azul y rojo las subsecuencias común y repetida, respectivamente, más largas encontradas. En este caso, el cálculo final de MLRCS utilizaría la subsecuencia común 'CUUGCU' por ser la de mayor longitud.

$$HD(CDS_i, CDS_j) = \sum_{1 \le k \le L} \sigma(CDS_{i,k}, CDS_{j,k}), \tag{3}$$

donde L es la longitud de una CDS en nucleótidos y $\sigma(\cdot)$ toma el valor 1 si los nucleótidos comparados $CDS_{i,k}$ y $CDS_{j,k}$ (posición k en cada CDS) son diferentes y 0 en caso contrario.

Como en el caso de la función anterior, tras calcular el valor de HD para todos los pares de CDSs, el objetivo vuelve a ser maximizar el valor mínimo de HD encontrado, como muestra la Ecuación 4:

$$mHD = \min_{1 \le i < j \le I} \quad \frac{HD(CDS_i, CDS_j)}{L}. \tag{4}$$

La tercera función objetivo (MLRCS) está centrada también en crear diferencias entre CDSs con el propósito de evitar la recombinación homóloga. En esta ocasión, el objetivo es romper la subsecuencia más larga repetida dentro de una misma CDS o común entre 2 CDSs. La búsqueda de esta subsecuencia se realiza según la Ecuación 5:

$$LRCS(CDS_i, CDS_j) = \underset{1 \le p,q,l \le L}{length(S_{i,p,l})} \quad when\ (S_{i,p,l} = S_{j,q,l}), \tag{5}$$

donde l es la longitud de la subsecuencia, y es posible que $i = j$, es decir, buscar en la misma CDS para encontrar subsecuencias repetidas con diferente posición de inicio ($p \neq q$). En la Figura 1 se muestra un ejemplo de este cálculo. El objetivo es minimizar la máxima longitud de subsecuencia repetida o común encontrada, como indica la Ecuación 6:

$$MLRCS = \max_{1 \le i \le j \le I} \quad \frac{LRCS(CDS_i, CDS_j)}{L}. \tag{6}$$

III. Algoritmo mutiobjetivo SFLA

El algoritmo SFLA (*Shuffled Frog Leaping Algorithm* [15]) es un método de optimización memético basado en una población de ranas (soluciones) que simula su comportamiento inteligente. El algoritmo comienza dividiendo la población en m grupos: la mejor solución al grupo 1, la segunda mejor al grupo 2, la m mejor solución al grupo m, la m+1 mejor solución al grupo 1 de nuevo, y así sucesivamente. Siendo el número de grupos un parámetro a establecer previamente. Con

la población dividida y todos los grupos con el mismo número de soluciones, cada uno de los grupos intenta mejorar hasta alcanzar un criterio de parada. Entonces, la población vuelve a reagruparse y, tras una nueva ordenación por calidad de todas las soluciones, vuelven a repartirse en grupos. Este mecanismo de dividir en grupos, mejorar localmente y volver a juntar la población es repetido hasta alcanzar el criterio de parada del algoritmo, combinando así la búsqueda local y global.

En este trabajo, proponemos una adaptación multiobjetivo del algoritmo SFLA (MOSFLA, *Multi-Objective SFLA*) para la codificación de proteínas con múltiples copias. El Algoritmo 1 describe el pseudocódigo de MOSFLA.

Algoritmo 1 Pseudocódigo del algoritmo MOSFLA.

Input: pop_size (número de soluciones o tamaño de la población), m (número de grupos), max_cycles (número máximo de generaciones), P_m (probabilidad de mutación)

Output: $nondom\_file$ (archivo con las soluciones no dominadas)

1: $nondom\_file \leftarrow \emptyset$;
2: $pop \leftarrow init\_population(pop\_size)$;
3: $pop \leftarrow order\_by\_R\&C(pop, pop\_size)$;
4: **for** $cycle \leftarrow 1, max\_cycles$ **do**
5: $groups \leftarrow divide\_pop(pop, pop\_size, m)$;
6: $groups \leftarrow improve\_groups(groups, m, P_m)$;
7: $pop \leftarrow merge(groups)$;
8: $pop \leftarrow order\_by\_R\&C(pop, 2 * pop\_size)$;
9: $nondom\_file \leftarrow save\_solutions(pop)$;
10: **end for**

En primer lugar (línea 1), se inicializa un archivo vacío para almacenar las soluciones no dominadas encontradas por el algoritmo. Tras esto, se inicializa la población con pop_size soluciones (línea 2). Todas las soluciones son inicializadas aleatoriamente, excepto una que es creada de forma avariciosa usando los codones sinónimos mejor adaptados, es decir, es una solución con mCAI igual a 1 y todas las CDSs idénticas.

Para poder realizar la primera división de la población, las soluciones son evaluadas según las 3 funciones objetivo (mCAI, mHD y MLRCS) y ordenadas utilizando los operadores multiobjetivo de rank y crowding [16] (línea 3), quedando las mejores soluciones en los primeros puestos. A continuación con la población ordenada comienza un bucle hasta alcanzar el criterio de parada definido, max_cycles generaciones en nuestro caso (línea 4). En cada iteración se lleva a cabo el proceso explicado anteriormente: la población es dividida en m grupos con la misma cantidad de soluciones cada uno (línea 5), individualmente cada grupo intenta mejorar su subpoblación llevando a cabo una búsqueda local (línea 6) y, por último, la población es reagrupada (línea 7). El proceso de mejora de los grupos genera una población hija a partir de la población padre, es decir, el tamaño de la población se duplica, por eso, se vuelve a ordenar por rank y crowding (línea 8) para que en la siguiente iteración del bucle otra vez se parta de las pop_size mejores soluciones. Al final de cada iteración las soluciones no dominadas se almacenan en $nondom\_file$ (línea 9).

Durante el proceso de mejora de cada grupo, se ejecuta un bucle cuyo número máximo de iteraciones coincide con el número de soluciones en el grupo. En cada iteración se intenta mejorar a la peor solución local del grupo. Para ello, se aplica el operador de mutación sobre la mejor solución local del grupo. Si la solución mutada no domina a la peor del grupo, se aplica el operador de mutación sobre la mejor solución global de la población completa. Si la solución mutada no domina a la peor del grupo, se genera una solución aleatoria. En cualquier caso, este proceso generará una solución hija (mutada o generada aleatoriamente).

El operador de mutación está compuesto por 4 tipos de mutación, aunque en cada uso del operador solamente un tipo, seleccionado de forma aleatoria, es aplicado. Tres de estos tipos de mutación se centran en un objetivo en particular y el cuarto actúa sobre la solución completa de forma aleatoria. Todas las mutaciones usan la probabilidad de mutación (P_m):

1. Cada codón de la CDS con el valor mínimo de CAI es cambiado por un sinónimo con mejor peso con una probabilidad P_m. Si no es posible cambiar por un sinónimo con mejor peso no se realiza el cambio.
2. Cada codón de la primera CDS del par de CDSs con el mínimo valor de HD es cambiado por otro sinónimo aleatorio con una probabilidad P_m.
3. Cada codón que forma parte de la subsecuencia repetida o común más larga es cambiado por otro sinónimo aleatorio con una probabilidad P_m.
4. Cada codón de cada CDS es reemplazado por otro sinónimo aleatorio con una probabilidad P_m.

En este trabajo hemos realizado una implementación en C/C++ del algoritmo MOSFLA.

IV. Experimentos y resultados

IV-A. *Configuración*

Para llevar a cabo los experimentos se han utilizado 9 proteínas reales (obtenidas de UniProt, *Universal Protein Resource*). Se han utilizado instancias muy variadas conforme a dos características que influyen directamente en la complejidad del problema: número de CDSs y longitud en aminoácidos (AA). La Tabla I presenta las 9 proteínas usadas.

Tabla I
Lista de proteínas reales utilizadas en los experimentos.

Código	Nombre	CDSs	Long (AA)	CDSs*Long
Q5VZP5	DUS27_HUMAN	2	1158	2316
A4Y1B6	FADB_SHEPC	3	716	2148
B3LS90	OCA5_YEAS1	4	679	2716
B4TWR7	CAIT_SALSV	5	505	2525
Q91X51	GORS1_MOUSE	6	446	2676
Q89BP2	DAPE_BRADU	7	388	2716
A6L9J9	TRPF_PARD8	8	221	1768
Q88X33	Y1415_LACPL	9	114	1026
B7KHU9	PETG_CYAP7	10	38	380

Con el objetivo de realizar una comparación justa con el trabajo encontrado en la literatura [14] (basado en NSGA-II),

Tabla II
VALORES DE CONFIGURACIÓN PROBADOS. EN NEGRITA EL MEJOR.

	Valores probados						
P_m	0,3125 %	0,625 %	**1,25** %	2,5 %	5 %	10 %	20 %
m	2	5	**10**	20			

el tamaño de la población y el número de generaciones se han establecido con los mismos valores: 100 soluciones en la población y 100 generaciones. El resto de parámetros para el método de Terai [14] se han establecido con los valores por defecto propuestos por sus autores. En el caso del algoritmo MOSFLA, la Tabla II muestra los dos parámetros adicionales que posee, así como los valores que se han probado para encontrar la mejor configuración: probabilidad de mutación (P_m) en 1,25 % y número de grupos (m) en 10.

Se han realizado 31 ejecuciones independientes de cada experimento para asegurar la fiabilidad estadística de todos los resultados. Además, los tres objetivos han sido normalizados en el rango [0,1], utilizando los valores ideal y nadir de la Tabla III.

Tabla III
VALORES IDEAL Y NADIR USADOS PARA NORMALIZAR LOS OBJETIVOS.

Objetivo	Valor nadir	Valor ideal
mCAI	0	1
mHD	0	0,40
MLRCS	1	0

IV-B. Redefinición del problema

Durante los primeros resultados obtenidos en los experimentos con MOSFLA encontramos que no existe conflicto entre los objetivos mHD y MLRCS: la mejora de uno de ellos no implica empeorar el otro, es decir, al contrario de lo esperado, son objetivos correlacionados y mejorar uno implica mejorar el otro. Esto no ocurre con mCAI, que al ser mejorado, mHD y MLRCS empeoran y viceversa.

Por este motivo, a diferencia de lo propuesto en la literatura científica [14], replanteamos el problema como una optimización de 2 objetivos. Para ello, es necesario verificar la calidad de las soluciones al optimizar 2 objetivos frente a la calidad que se obtiene si optimizamos 3 objetivos, y además, comprobar cuál de los objetivos, mHD o MLRCS, puede dejar de optimizarse sin que la calidad de las soluciones se vea afectada. La optimización de mCAI está asegurada por ser conflictivo con el resto de objetivos.

Para comprobar la calidad de los frentes de Pareto obtenidos, una métrica ampliamente utilizada en optimización multiobjetivo es el hipervolumen (HV) [17] o medida de Lebesgue (Leb). Se trata de una métrica unaria que mide el porcentaje del espacio objetivo que es cubierto por las soluciones que forman el frente de Pareto $\mathcal{A}$, y es calculado según la Ecuación 7:

$$HV(\mathcal{A}, r) = Leb\left(\bigcup_{i=1}^{|\mathcal{A}|} h(a_i, r)\right), \qquad (7)$$

Tabla IV
RESULTADOS DEL HIPERVOLUMEN (MEDIANA$_{\pm DESVIACION_CUARTILICA}$) PARA LAS 3 VARIANTES DE MOSFLA. EN NEGRITA EL MEJOR RESULTADO.

Proteína	MOSFLA	MOSFLA-2CH	MOSFLA-2CL
Q5VZP5	64,48 %$_{\pm 0,72\,\%}$	**65,77 %**$_{\pm 0,36\,\%}$	40,83 %$_{\pm 1,05\,\%}$
A4Y1B6	53,56 %$_{\pm 0,19\,\%}$	**53,81 %**$_{\pm 0,25\,\%}$	35,29 %$_{\pm 2,62\,\%}$
B3LS90	56,18 %$_{\pm 0,22\,\%}$	**56,61 %**$_{\pm 0,23\,\%}$	40,47 %$_{\pm 1,16\,\%}$
B4TWR7	50,13 %$_{\pm 0,26\,\%}$	**50,30 %**$_{\pm 0,14\,\%}$	36,27 %$_{\pm 0,80\,\%}$
Q91X51	52,37 %$_{\pm 0,16\,\%}$	**52,39 %**$_{\pm 0,16\,\%}$	40,47 %$_{\pm 0,64\,\%}$
Q89BP2	50,29 %$_{\pm 0,13\,\%}$	**50,32 %**$_{\pm 0,16\,\%}$	39,41 %$_{\pm 0,65\,\%}$
A6L9J9	46,88 %$_{\pm 0,14\,\%}$	**47,06 %**$_{\pm 0,17\,\%}$	36,42 %$_{\pm 0,67\,\%}$
Q88X33	**42,55 %**$_{\pm 0,27\,\%}$	42,48 %$_{\pm 0,28\,\%}$	31,61 %$_{\pm 1,65\,\%}$
B7KHU9	**40,61 %**$_{\pm 0,33\,\%}$	40,01 %$_{\pm 0,57\,\%}$	31,29 %$_{\pm 1,47\,\%}$
Media	50,78 %	**50,97 %**	36,89 %

donde $|\mathcal{A}|$ es la cardinalidad de $\mathcal{A}$ y $h(a_i, r)$ representa el volumen que define cada punto a_i del frente de Pareto $\mathcal{A}$ con respecto al punto de referencia r.

Después de calcular esta métrica para las 3 variantes, denominadas como MOSFLA para la optimización con 3 objetivos, MOSFLA-2CH para la optimización de mCAI y mHD, y MOSFLA-2CL para la optimización de mCAI y MLRCS, la Tabla IV muestra los resultados obtenidos. Para que la comparación sea correcta, en las variantes donde solo se optimizan 2 objetivos, el hipervolumen se calcula considerando los 3 objetivos (aunque uno de ellos no se optimice). Los resultados revelan que la variante MOSFLA-2CH consigue una amplia victoria en todas las instancias frente a MOSFLA-2CL y, además, también mejora ligeramente en 7 de las 9 instancias si se compara con la variante MOSFLA.

Por lo tanto, podemos concluir que la optimización de los 2 objetivos mCAI y mHD implica indirectamente la optimización del tercero de los objetivos definidos (MLRCS). En consecuencia, para el resto de comparativas de este artículo utilizaremos la variante MOSFLA-2CH.

IV-C. Comparativa con el estado del arte

Como se ha mencionado, MOSFLA-2CH será comparado con el trabajo encontrado en la literatura [14] centrado en la optimización de los 3 objetivos. La configuración utilizada para MOSFLA-2CH es la misma indicada anteriormente para MOSFLA, con la excepción de P_m que obtiene mejores resultados con el valor 0,625 %. La comparativa se realizará utilizando 3 métricas de calidad distintas: hipervolumen, cobertura del conjunto [18] y máxima extensión [19]. Para que la comparación sea justa, aunque MOSFLA-2CH solo optimiza 2 objetivos, todas las métricas de calidad se han calculado de igual forma para ambos métodos (MOSFLA-2CH y Terai [14]): utilizando los 3 objetivos.

IV-C1. Hipervolumen: El hipervolumen ha sido calculado como se explicó anteriormente. La Tabla V muestra la mediana junto a la desviación cuartílica del hipervolumen obtenido por ambos métodos: MOSFLA-2CH y Terai [14]. En todas las instancias MOSFLA-2CH obtiene mejor hipervolumen.

Para garantizar que las diferencias entre ambos métodos son estadísticamente significativas se aplica el test paramétrico

Tabla V
COMPARATIVA DEL HIPERVOLUMEN (MEDIANA$_{\pm DESVIACION_CUARTILICA}$). EN NEGRITA EL MEJOR RESULTADO.

Proteína	MOSFLA-2CH	Método Terai [14]	¿Significancia estadística?
Q5VZP5	**64,49 %**$_{\pm 0,47\,\%}$	59,92 %$_{\pm 0,18\,\%}$	Sí
A4Y1B6	**54,74 %**$_{\pm 0,27\,\%}$	52,53 %$_{\pm 0,06\,\%}$	Sí
B3LS90	**57,22 %**$_{\pm 0,19\,\%}$	54,62 %$_{\pm 0,16\,\%}$	Sí
B4TWR7	**50,54 %**$_{\pm 0,14\,\%}$	48,91 %$_{\pm 0,21\,\%}$	Sí
Q91X51	**52,73 %**$_{\pm 0,17\,\%}$	50,47 %$_{\pm 0,23\,\%}$	Sí
Q89BP2	**50,61 %**$_{\pm 0,22\,\%}$	48,61 %$_{\pm 0,22\,\%}$	Sí
A6L9J9	**47,06 %**$_{\pm 0,22\,\%}$	45,56 %$_{\pm 0,13\,\%}$	Sí
Q88X33	**42,32 %**$_{\pm 0,25\,\%}$	41,07 %$_{\pm 0,09\,\%}$	Sí
B7KHU9	**38,89 %**$_{\pm 0,45\,\%}$	38,26 %$_{\pm 0,12\,\%}$	Sí
Media	**50,96 %**	48,88 %	

Tabla VI
COMPARATIVA DE LA COBERTURA DEL CONJUNTO. EN NEGRITA EL MEJOR RESULTADO.

Proteína	SC(MOSFLA-2CH, Método Terai [14])	SC(Método Terai [14], MOSFLA-2CH)
Q5VZP5	**19,00 %**	8,43 %
A4Y1B6	**27,00 %**	4,36 %
B3LS90	**14,00 %**	1,02 %
B4TWR7	**10,00 %**	6,47 %
Q91X51	**5,00 %**	4,83 %
Q89BP2	**11,00 %**	5,83 %
A6L9J9	5,00 %	**6,83 %**
Q88X33	4,00 %	**7,83 %**
B7KHU9	**32,00 %**	8,83 %
Media	**14,11 %**	6,05 %

ANOVA si las muestras siguen una distribución normal y sus varianzas son homogéneas. En caso de no cumplir estas condiciones, se aplica el test no paramétrico U de Mann-Whitney. Todos los test estadísticos se realizan con un nivel de confianza de 95 % (p-valor=0,05). Los resultados del análisis estadístico se muestran en la cuarta columna de la Tabla V. Para las 9 instancias se puede asegurar con un 95 % de confianza que las diferencias de hipervolumen obtenidas entre ambos métodos son estadísticamente significativas.

IV-C2. Cobertura del conjunto: La cobertura del conjunto (SC, *Set Coverage* [18]) es una métrica binaria que compara la calidad de un frente de Pareto respecto a otro. Dado dos frentes de Pareto $\mathcal{A}$ y $\mathcal{B}$, SC($\mathcal{A}$, $\mathcal{B}$) calcula el porcentaje de soluciones del frente de Pareto $\mathcal{B}$ que son cubiertas (dominadas o iguales, $\succeq$) por, al menos, una solución del frente de Pareto $\mathcal{A}$, como muestra la Ecuación 8:

$$SC(\mathcal{A},\mathcal{B}) = \frac{|\{b \in \mathcal{B}; \exists\ a \in \mathcal{A} : a \succeq b\}|}{|\mathcal{B}|}, \qquad (8)$$

donde $|\mathcal{B}|$ es la cardinalidad de $\mathcal{B}$.

La cobertura del conjunto no es una métrica simétrica, por lo que es necesario calcularla en ambas direcciones SC($\mathcal{A}$, $\mathcal{B}$) y SC($\mathcal{B}$, $\mathcal{A}$). Para realizar el cálculo de esta métrica se han empleado los frentes de Pareto con hipervolumen mediana de cada método. La Tabla VI muestra los resultados de esta comparativa entre MOSFLA-2CH y Terai [14]. Para 7 de las 9 instancias, la mayoría, MOSFLA-2CH consigue mejores resultados.

En términos generales, el valor medio obtenido por MOSFLA-2CH es de un 14,11 % mientras que el método de Terai solo llega a un 6,05 %. Por lo tanto, podemos concluir que los frentes de Pareto producidos por MOSFLA-2CH, según la cobertura del conjunto, son de mejor calidad que los obtenidos por el método de Terai [14].

IV-C3. Máxima extensión: La máxima extensión (MS, *Maximum Spread* [19]) es una métrica unaria que mide la extensión del frente de Pareto $\mathcal{A}$ en el espacio objetivo. Se trata de una métrica que basa su cálculo en la distancia euclídea del valor máximo y mínimo de cada objetivo obtenidos por las soluciones de dicho frente de Pareto, como muestra la Ecuación 9:

$$MS(\mathcal{A}) = \sqrt{\sum_{m=1}^{M} (max_{i=1}^{NDS} f_m^i - min_{i=1}^{NDS} f_m^i)^2}, \qquad (9)$$

donde M es el número de objetivos y NDS indica el número de soluciones no dominadas que forman el frente de Pareto $\mathcal{A}$.

La Tabla VII muestra la máxima extensión mediana junto a su desviación cuartílica, obtenidas por MOSFLA-2CH y Terai [14]. En 8 de las 9 instancias, casi la totalidad, MOSFLA-2CH consigue mejores resultados, lo que indica que los frentes de Pareto de MOSFLA-2CH son más extensos y entre sus extremos hay una mayor distancia.

Tabla VII
COMPARATIVA DE LA MÁXIMA EXTENSIÓN (MEDIANA$_{\pm DESVIACION_CUARTILICA}$). EN NEGRITA EL MEJOR RESULTADO.

Proteína	MOSFLA-2CH	Método Terai [14]	¿Significancia estadística?
Q5VZP5	**1,691**$_{\pm 0,004}$	1,349$_{\pm 0,005}$	Sí
A4Y1B6	**1,292**$_{\pm 0,003}$	1,284$_{\pm 0,001}$	Sí
B3LS90	**1,285**$_{\pm 0,002}$	1,278$_{\pm 0,002}$	Sí
B4TWR7	1,263$_{\pm 0,003}$	**1,264**$_{\pm 0,001}$	No
Q91X51	**1,314**$_{\pm 0,003}$	1,309$_{\pm 0,001}$	Sí
Q89BP2	**1,290**$_{\pm 0,002}$	1,286$_{\pm 0,003}$	Sí
A6L9J9	**1,249**$_{\pm 0,003}$	1,242$_{\pm 0,003}$	Sí
Q88X33	**1,162**$_{\pm 0,004}$	1,158$_{\pm 0,004}$	No
B7KHU9	**1,199**$_{\pm 0,006}$	1,184$_{\pm 0,005}$	Sí
Media	**1,305**	1,262	

Para asegurar que estas diferencias son estadísticamente significativas se ha realizado un análisis estadístico de la misma forma que se hizo para el hipervolumen. Los resultados, mostrados en la cuarta columna de la Tabla VII, indican que la instancia B4TWR7, donde el método de Terai [14] consigue mejor resultado que MOSFLA-2CH, no presenta una diferencia significativa estadísticamente. Sin embargo, en 7 de las 8 instancias restantes, donde MOSFLA-2CH obtiene mejor

resultado, se confirma que estadísticamente las mejoras son significativas.

V. Conclusiones y trabajos futuros

En este trabajo se aborda uno de los problemas que plantea la biología sintética, que es codificar una proteína con múltiples copias, como un problema de optimización multiobjetivo. La propuesta que realiza la literatura reciente está basada en la optimización de 3 objetivos focalizados sobre dos aspectos biológicos. Por un lado, el primer objetivo, mCAI, trata de optimizar el uso de los codones sinónimos mejor adaptados. Por otro lado, se definen dos objetivos, mHD y MLRCS, para crear diferencias entre las copias que componen la codificación de una proteína, evitando así la recombinación homóloga. La primera aportación de este trabajo es la redefinición del problema al encontrar que los objetivos mHD y MLRCS no son conflictivos entre sí, sino que están correlacionados. En particular, después de realizar los primeros experimentos, concluimos que la optimización de mHD implica la optimización de MLRCS de forma indirecta.

Para este problema de optimización, hemos realizado una implementación multiobjetivo del algoritmo SFLA. Se trata de un algoritmo que aplica una metaheurística memética basada en el comportamiento inteligente de una población de ranas (soluciones). Con el propósito de realizar una comparativa entre la literatura relacionada y nuestra propuesta, se han realizado experimentos con 9 proteínas reales (considerando escenarios muy distintos) y los resultados se han analizado con tres métricas de calidad diferentes: hipervolumen, cobertura del conjunto y máxima extensión.

En los experimentos iniciales se confirma la correlación entre los objetivos mHD y MLRCS, y que la optimización de mCAI junto a mHD alcanza la misma calidad en la métrica del hipervolumen que optimizando los 3 objetivos. Por esta razón, para el resto de la fase experimental se utiliza la implementación denomina MOSFLA-2CH con solo 2 objetivos de optimización (mCAI y mHD). En la comparativa de nuestra propuesta MOSFLA-2CH frente al método de Terai [14], como trabajo relacionado encontrado en la literatura, los resultados indican la mejora clara de MOSFLA-2CH en todas las métricas de calidad utilizadas, y además estas diferencias son confirmadas por análisis de significancia estadística.

Como trabajo futuro proponemos comparar estos resultados con otras implementaciones de algoritmos multiobjetivo y poder así ratificar la calidad de los resultados obtenidos por la adaptación realizada del algoritmo SFLA. Por otro lado, viendo los buenos resultados del algoritmo MOSFLA, proponemos utilizar este algoritmo en problemas de optimización multiobjetivo de otros ámbitos.

Agradecimientos

Esta investigación ha sido parcialmente financiada por el Ministerio de Ciencia, Innovación y Universidades y la Agencia Estatal de Investigación (proyecto PID2019-107299GB-I00/AEI/10.13039/501100011033), por la Junta de Extremadura (ayuda GR18090) y por la Unión Europea (Fondo Europeo de Desarrollo Regional).

Referencias

[1] P. Gu, F. Yang, T. Su, Q. Wang, Q. Liang, and Q. Qi, "A rapid and reliable strategy for chromosomal integration of gene(s) with multiple copies," *Scientific Reports*, vol. 5, p. 9684, Apr 2015.

[2] K. E. J. Tyo, P. K. Ajikumar, and G. Stephanopoulos, "Stabilized gene duplication enables long-term selection-free heterologous pathway expression," *Nature Biotechnology*, vol. 27, pp. 760–765, Jul 2009.

[3] A. Vassileva, D. A. Chugh, S. Swaminathan, and N. Khanna, "Expression of hepatitis B surface antigen in the methylotrophic yeast Pichia pastoris using the GAP promoter," *Journal of Biotechnology*, vol. 88, no. 1, pp. 21–35, 2001.

[4] H. Hohenblum, B. Gasser, M. Maurer, N. Borth, and D. Mattanovich, "Effects of gene dosage, promoters, and substrates on unfolded protein stress of recombinant Pichia pastoris," *Biotechnology and Bioengineering*, vol. 85, no. 4, pp. 367–375, 2004.

[5] R. Aw and K. M. Polizzi, "Can too many copies spoil the broth?" *Microbial Cell Factories*, vol. 12, no. 1, p. 128, Dec 2013.

[6] F. K. Khasanov, D. J. Zvingila, A. A. Zainullin, A. A. Prozorov, and V. I. Bashkirov, "Homologous recombination between plasmid and chromosomal DNA in Bacillus subtilis requires approximately 70 bp of homology," *Molecular and General Genetics*, vol. 234, no. 3, pp. 494–497, Sep 1992.

[7] P. Manivasakam, S. C. Weber, J. McElver, and R. H. Schiestl, "Micro-homology mediated PCR targeting in Saccharomyces cerevisiae," *Nucleic Acids Research*, vol. 23, no. 14, pp. 2799–2800, Jul 1995.

[8] J. Athey, A. Alexaki, E. Osipova, A. Rostovtsev, L. V. Santana-Quintero, U. Katneni, V. Simonyan, and C. Kimchi-Sarfaty, "A new and updated resource for codon usage tables," *BMC Bioinformatics*, vol. 18, no. 1, p. 391, Sep 2017.

[9] G. R. Webster, A. Y.-H. Teh, and J. K.-C. Ma, "Synthetic gene design - The rationale for codon optimization and implications for molecular pharming in plants," *Biotechnology and Bioengineering*, vol. 114, no. 3, pp. 492–502, 2017.

[10] C.-H. Yu, Y. Dang, Z. Zhou, C. Wu, F. Zhao, M. S. Sachs, and Y. Liu, "Codon usage influences the local rate of translation elongation to regulate co-translational protein folding," *Molecular Cell*, vol. 59, no. 5, pp. 744–754, 2015.

[11] T.-A. Tran, N. T. Vo, H. D. Nguyen, and B. T. Pham, "A novel method to predict highly expressed genes based on radius clustering and relative synonymous codon usage," *Journal of Computational Biology*, vol. 22, no. 12, pp. 1086–1096, Dec 2015.

[12] J. X. Chin, B. K.-S. Chung, and D.-Y. Lee, "Codon Optimization OnLine (COOL): a web-based multi-objective optimization platform for synthetic gene design," *Bioinformatics*, vol. 30, no. 15, pp. 2210–2212, 2014.

[13] J. C. Guimaraes, M. Rocha, A. P. Arkin, and G. Cambray, "D-Tailor: automated analysis and design of DNA sequences," *Bioinformatics*, vol. 30, no. 8, pp. 1087–1094, 2014.

[14] G. Terai, S. Kamegai, A. Taneda, and K. Asai, "Evolutionary design of multiple genes encoding the same protein," *Bioinformatics*, vol. 33, no. 11, pp. 1613–1620, Jun 2017.

[15] E. Elbeltagi, T. Hegazy, and D. Grierson, "A modified shuffled frog-leaping optimization algorithm: applications to project management," *Structure and Infrastructure Engineering*, vol. 3, no. 1, pp. 53–60, 2007.

[16] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, Apr 2002.

[17] N. Beume, C. M. Fonseca, M. Lopez-Ibanez, L. Paquete, and J. Vahrenhold, "On the complexity of computing the hypervolume indicator," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 5, pp. 1075–1082, Oct 2009.

[18] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. G. da Fonseca, "Performance assessment of multiobjective optimizers: an analysis and review," *IEEE Transactions on Evolutionary Computation*, vol. 7, no. 2, pp. 117–132, Apr 2003.

[19] E. Zitzler, K. Deb, and L. Thiele, "Comparison of multiobjective evolutionary algorithms: empirical results," *Evolutionary Computation*, vol. 8, no. 2, pp. 173–195, Jun 2000.

Recognition of Teaching Activities from University Lecture Transcriptions*

*Note: The full contents of this paper have been published in the volume Lecture Notes in Artificial Intelligence 12882 (LNAI 12882)

Daniel Diosdado, Alberto Romero, Eva Onaindia
VRAIN
Universitat Politcnica de Valncia
Valencia, Spain
{dadiolo, alrofer}@inf.upv.es, onaindia@dsic.upv.es

***Abstract*—Research on language acquisition for academic purposes is not extensive. In this work, we propose to build a system for recognizing teaching activities from automatic transcriptions of classroom audio and video recordings centered on the professor's discourse. To this end, we identified the main teaching activities that cover the nature of the lecturer discourse when giving a course eg. 'theoretical explanation', real-world practical example', interaction lecturer-student', 'course-related asides', etc. We labeled a dataset of lecture transcriptions from a repository with an approximate length of 50 hours and we build a classifier by fine-tuning the XLM-RoBERTa model with a classification head on top of it. The results will show that our proposal is a promising step ahead towards recognition of discourse activities in academic contexts.**

***Index Terms*—Spoken academic lecture, Text classification, Transformer models**

DOI: https://doi.org/10.1007/978-3-030-85713-4_22

Human Activity Recognition with Capsule Networks*

*Note: The full contents of this paper have been published in the volume Lecture Notes in Artificial Intelligence 12882 (LNAI 12882)

Laura Llopis-Ibor, Alfredo Cuesta-Infante, César Beltrán-Royo, Juan José Pantrigo
Universidad Rey Juan Carlos
Móstoles, Spain
{laura.llopis, alfredo.cuesta, cesar.beltran, juanjose.pantrigo}@urjc.es

***Abstract*—Human activity recognition is a challenging problem, where deep learning methods are showing to be very efficient. In this paper we propose the use of capsule networks. This type of networks have proved to generalize better to novel viewpoints than convolutional neural networks. We show that the use of capsule networks into a straightforward architecture, between a convolutional preprocessing stage to extract visual features and a header for carrying out the task, is able to attain competitive results with spatio-temporal data without the use of any kind of recurrent neural network. Moreover, an analysis of the obtained results shows that our architecture is capable of learning the properties that encode the spatio-temporal dynamics of the movements that characterize each activity.**

***Index Terms*—Capsule Network, Human Activity Recognition, Skeleton Based Action Recognition, Image Embedding**

DOI: https://doi.org/10.1007/978-3-030-85713-4_8

Identification of Contact Failures in Multilayered Media via Inverse Problem and Artificial Neural Networks

L.C.S. Jardim, D.C. Knupp, R.P. Domingos,
L.A.S. Abreu, A.J. Silva Neto
Instituto Politécnico
Universidade do Estado do Rio de Janeiro
Nova Friburgo, Brazil
{ljardim, diegoknupp, roberto.pinheiro, luiz.abreu, ajsneto}@iprj.uerj.br

C. Cruz Corona
Dept. de Ciencias de la Computación
Universidad de Granada
Granada, Spain
carloscruz@decsai.ugr.es

***Abstract*—An Artificial Neural Network is used to estimate the positioning of different materials in multilayered media. The network is modeled as an Inverse Problem: it receives temperature measurements from the exposed surface of the media as input and yields estimates of the sought thermal properties. To test this approach, the problem of contact failures identification between two materials is proposed. Such problem consists of estimating the presence of air inclusions at the adhesive layer used to join two materials, which can be used to infer the quality of bonding in manufacturing process. Results show that this is a promising approach due to the good accuracy and low computational cost observed for the numerical examples here presented.**

***Index Terms*—Artificial Neural Networks, Inverse Heat Transfer Problem, Contact Failures Identification**

I. Introduction

In order to estimate thermal properties of materials, an Inverse Heat Conduction Problem (IHCP) may be formulated and solved. In classical approaches, IHCPs can be treated as optimization problems, where the squared difference between temperature measurements and predictions from a model must be minimized [1], [2], or as bayesian inference problems, such as the Maximum a Posteriori and the Markov Chain Monte Carlo methods [3], [4]. Since these types of IHCPs formulations require intensive iterative procedures, many efforts of the scientific community are focused on reducing the related computational costs [5], [6]. Parallel to these efforts, the use of computational intelligence emerge as a possible way of solving IHCPs, such as Genetic Algorithms [7], [8] and Artificial Neural Networks (ANN) [9]–[12], which are considered soft computing approaches, as opposed to classical ones.

In this work the IHCP of estimating the spatially varying thermal properties of a multilayered medium is solved via an ANN, where the input data are temperature measurements of the exposed surface of the medium and the output is a set of discrete points representing the thermal properties of different materials. More specifically, in order to test this approach, the problem of contact failures identification was chosen, where two materials are joined with an adhesive layer. Gaps of air can form in this layer, meaning that the adhesion process is not perfect [13]–[15]. Identifying the thermal conductance of this layer containing gaps is crucial to access the adhesion quality in manufacturing.

The network training is performed with multiple contact failure configurations where the input are simulated experimental data, and they are drawn from the solution of the direct problem. Then, the input-output training pairs are constructed. After testing multiple ANN configurations, the ANN that generated the smaller error when compared to validation data is selected. Finally, the results show that this approach is promising and computationally efficient for the contact failure identification problem, specially after the training phase.

This work was funded by CAPES/Brazil, CNPq/Brazil, FAPERJ/Brazil and Spain

II. Direct Problem Formulation and Solution

Consider the $L_x \times L_y$ multilayered rectangular medium presented in Fig. 1. Both left and right boundaries are thermally insulated, the constant heat flux q is prescribed at the bottom boundary, and the top one is considered exposed, with ambient temperature of T_∞ and natural convection heat transfer coefficient h. Temperature measurements $\boldsymbol{T}^{\text{exp}}$ are acquired at the exposed boundary, i.e. $y = L_y$. Illustratively, such medium is represented in Fig. 1 containing three different layers and two gaps in the middle layer.

Using a single-domain formulation and taking in consideration the mentioned assumptions, the heat conduction problem can be written as

$$\frac{\partial}{\partial x}\left(k(x,y)\frac{\partial T(x,y,t)}{\partial x}\right)+\frac{\partial}{\partial y}\left(k(x,y)\frac{\partial T(x,y,t)}{\partial x}\right)= \rho(x,y)c_p(x,y)\frac{\partial T(x,y,t)}{\partial t}, \tag{1a}$$

$$\left.\frac{\partial T(x,y,t)}{\partial x}\right|_{x=0}=0,\quad \left.\frac{\partial T(x,y,t)}{\partial x}\right|_{x=L_x}=0, \tag{1b,c}$$

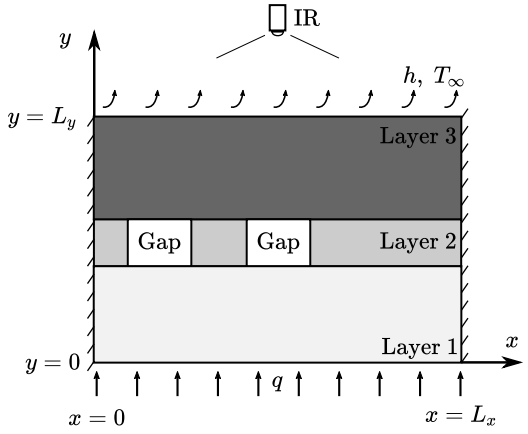


Fig. 1. Schematic representation of a multilayered medium.

$$-k(x,y)\left.\frac{\partial T(x,y,t)}{\partial y}\right|_{y=0} = q, \tag{1d,e}$$

$$k(x,y)\left.\frac{\partial T(x,y,t)}{\partial y}\right|_{y=L_y} + hT(x,L_y,t) = hT_\infty,$$

$$T(x,y,0) = T_\infty, \tag{1f}$$

where h is the convection heat transfer coefficient at the top boundary, and q is the prescribed heat flux at the bottom boundary. The materials properties $k(x,y), \rho(x,y)$, and $c_p(x,y)$ are the thermal conductivity, specific mass, and specific heat, respectively. These spatially dependent properties are modeled in such way to represent the different materials of the multilayered medium. Furthermore, the initial temperature is considered equal to the ambient temperature T_∞.

The problem described by Eq. (1) is solved using the built-in "NDSolve" routine of the Wolfram Mathematica 11.0 software. This routine is executed with the Finite Elements option and the mesh is generated with rectangular user defined uniform cell size. This routine returns an interpolated temperature profile $T(x,y,t)$ for the whole $L_x \times L_y$ medium and for times ranging from $t = 0$ to $t = t_{\max}$, which is a prescribed maximum time instant.

III. Inverse Problem Formulation and Solution via Artificial Neural Networks

Consider that the materials of the multilayered medium presented in Fig. 1 are known, but the position of the gaps in Layer 2 are unknown. In this case, the inverse problem is formulated based on the assumption that Layer 2 is composed either by its own material or the gap's material. This leads to the definition of an auxiliary function $\gamma(x)$ which represents the thermal properties $k(x,y)$ and $\rho(x,y)c_p(x,y)$ of Layer 2, defined as

$$\gamma(x) = \frac{\phi(x) - \phi_{\text{gap}}}{\phi_{\text{layer}} - \phi_{\text{gap}}}, \tag{2}$$

where $\phi(x)$ is the thermal conductivity $k(x,y)$ or the volumetric heat capacity $\rho(x,y)c_p(x,y)$ for all of Layer 2 materials, which only varies in the x coordinate, and the subscripts "gap" and "layer" represent such thermal properties of the gap or the layer's material. The function γ is used algorithmically as a way to translate the thermal properties to the spatial dependent boolean function "gap, no-gap", "0" or "1", respectively. Furthermore, γ can be reversely translated to the thermal properties $k(x,y)$ and $\rho(x,y)c_p(x,y)$ of Layer 2, since the values ϕ_{gap} and ϕ_{layer} are known.

In order to obtain estimates of the gaps positioning in Layer 2, an ANN is used to represent the inverse model, i.e. the network receives temperature experimental data as input and outputs a vector representing discrete points of the auxiliary function $\gamma(x)$. The experimental data is computationally simulated, as described in sub-section III-A, then, the construction of the neural network and the assumptions taken in consideration for generating the training data set are explained in sub-section III-B.

A. Experimental Data

In order to simulate real measurements of temperature, the experimental data is computationally generated, where random numbers r drawn from a normal distribution centered at zero with known standard deviation σ are added to solution of the direct problem, i.e.

$$T^{\text{exp}}_{i,j} = T(x_i, L_y, t_j) + r,\ r \sim \text{N}(0,\sigma^2),$$
$$i = 1,2,...,N_x,\ j = 1,2,...,N_t, \tag{3}$$

where x_i represents the position where the temperature are acquired along the x coordinate with the total number of N_x points, and t_j the time levels with total number of N_t instants. Therefore, the total number of experimental data considered is $N_m = N_x \times N_t$.

B. The Artificial Neural Network Inverse Model

As previously mentioned, the inverse problem of this work has the objective of estimating the positioning of the gaps that can occur in the Layer 2 of the medium (see Fig. 1). As presented in Fig. 2, the ANN construction starts with the temperature experimental data, then, these data are normalized between 0 and 1 and joined as only one vector. This normalization is performed with $T_{\min}$ and $T_{\max}$, which are the minimum and maximum temperature values from all the training data. Subsequently, this normalized data is passed as the input $\mathbf{E}$ of the network. Then $\mathbf{E}$ is passed to fully connected hidden layers, which are modified by weights, biases and activation functions. Finally, the output vector $\mathbf{G}$ is obtained as a representation of discrete points for the auxiliary function $\gamma(x)$.

The training data set consists of a list of ANN input-output pairs. They are constructed similarly to the experimental data in 3, by adding simulated noise to the solution of the direct problem 1. To build the training data set, it is important to follow some criteria that represents the physics of problem in order to avoid unnecessary computations [12]. In this work, the training data set is built by assuming that:

- There is a maximum number of gaps in Layer 2, i.e. $N_{\text{gap}} = 1,2,...,\text{MAX}_{\text{gap}}$;
- There is a minimum gap size, namely $L_{\min}$;

- The gaps only occur in the region of $x_{\min} \leq x \leq x_{\max}$, where $x_{\min}$ and $x_{\max}$ are chosen to avoid gaps being too close to the boundaries.

With these criteria, the thermal properties positioning represented by $\gamma(x)$ are randomly generated and plugged in the direct problem in order to generate the temperature profile necessary for the ANN's input.

Finally, two cases are appended to the training data set, the one where everything is a gap and the one that there is no gap. These cases represent the limits which the network must learn upon.

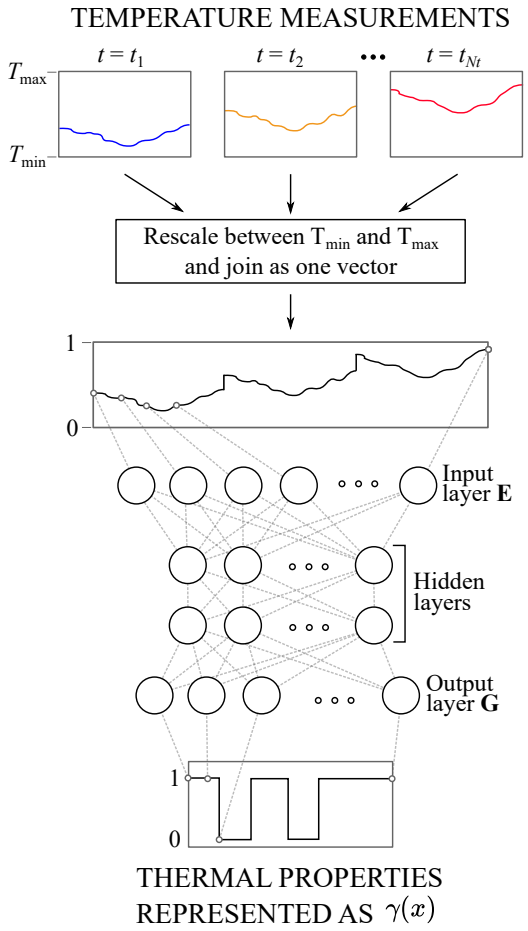


Fig. 2. Schematic representation of the ANN inverse model.

C. Testing and Validation

To determine the most accurate ANN configuration, a validation data set is built similarly to the training data set. This validation set is composed of N_{val} input-output pairs and they were never used during training. The comparison is performed by evaluating the expression

$$Q = \frac{1}{N_{\text{val}}} \sum_{i=1}^{N_{\text{val}}} ||\mathbf{G}_{\text{val}} - \mathbf{G}_{\text{net}}||_i^2 \tag{4}$$

where the operator $||.||$ is the l_2 Euclidean norm, $\mathbf{G}_{\text{val}}$ is the exact thermal properties of the i-th validation data in terms of $\gamma(x)$, and $\mathbf{G}_{\text{net}}$ is the obtained output of the ANN with the i-th validation noisy temperature input.

IV. Results and Discussion

Considering an steel plate for Layer 1, adhesive for Layer 2, composite protection for Layer 3 and air as the gap material, the respective thermal properties used for the solution of the direct problem are displayed in Tab. I.

TABLE I
THERMAL PROPERTIES FOR THE MULTILAYERED MEDIUM

Layer	k [W/mK]	ρc_p [MJ/m^{3}K]
1	13.400	3.860
2	0.700	1.750
3	1.171	2.650
Gap	0.0263	1.170

Furthermore, to obtain the results of the direct problem, the prescribed heat flux, the natural convection coefficient, and the ambient temperature are defined as $q = 10000$ W/m^2, $h = 15.0$ W/m^{2}K, and $T_\infty = 20.0$ °C, respectively. The medium dimensions are defined as $L_x = 0.2$ m and $L_y = 0.071$ m. Layer 2 is vertically positioned in the range $0.05 \leq y \leq 0.051$ m.

The experimental data are acquired at the exposed surface of the medium, with $N_x = 107$ evenly distributed points in x, with time instants of $t_1 = 1200$ s, $t_2 = 1500$ s, and $t_3 = 1800$ s, yielding a total of $N_t = 3$ number of instants and $N_m = 321$ total temperature data. Finally, they are corrupted with a simulated noise with standard deviation of 0.1 °C in 3. This type of measurement is typically obtained via infrared thermography techniques [16].

A. Testing and Validation Results

The network output **G** used consists of 201 evenly distributed points to represent the auxiliary function $\gamma(x)$, which leads to an interval of 0.001 m in x. Therefore the input **E** and output **G** sizes are 321 and 201, respectively.

All the training data are generated with 1, 2 or 3 gaps in Layer 2, i.e. $\text{MAX}_{\text{gap}} = 3$. For each number of gaps, 2000 randomly positioned gaps configurations are calculated and stored as the pair input-output, making the total size of the training set equal to 6002 - two cases containing the "no gap" and "only gap" configuration are appended to the set. Furthermore, since the input of the training data contains simulated noise generated by 3, the set of 6002 solutions are replicated three more times with new random noise generation, leading to a final training set of 24008 input-output pairs. This approach can increase the accuracy of the network in exchange of increased training time, since more examples are given. The minimum and maximum temperature values $T_{\min}$ and $T_{\max}$ are 26.96 °C and 60.56 °C, respectively. The training is performed with the ADAM optimizer, as implemented in the built-in routine "NetTrain" of the Wolfram Mathematica 11.0 software.

The validation data are assembled similarly to the training data, but with a size of 50 input-output pairs for the each

number of gaps. Therefore $N_{\text{val}} = 150$ data. Different ANN configuration are tested: with 1, 2 or 3 hidden layers, with 50, 100, 200, 300, 400 or 500 neurons per layer, and the ReLu or Tanh activation functions. The training is stopped when 6000 rounds with batch size of 128 is reached. The results obtained for the comparison of the obtained output G_{net} with the exact G_{val} with 4 are presented in Tab. II.

The best configuration, as presented in Tab. II, is the one with 3 hidden layers, 100 neurons per hidden layer and using the ReLU activation function. This configuration led to an error Q of 2.90796. Also, it is possible to observe that the results do not improve as the number of neurons per layer increases, which can be associated to an overfitting problem.

TABLE II
PERFORMANCE COMPARISON OF DIFFERENT ANN CONFIGURATIONS WITH THE VALIDATION DATA

Activation Function	Neurons per Layer	No. of Hidden Layers		
		1	2	3
Tanh	50	3.82877	3.05217	3.05969
	100	3.36145	3.00087	4.39504
	200	3.34842	3.78696	5.57865
	300	3.33625	4.61372	6.16197
	400	3.38737	5.08416	6.76938
	500	3.43878	5.54402	6.75651
ReLU	50	5.24817	4.15898	3.80961
	100	4.57393	3.17634	2.90796
	200	3.69342	2.96138	3.75685
	300	3.44163	3.10183	4.87376
	400	3.39889	3.30788	5.34571
	500	3.40509	3.76281	5.92278

B. Computational Experiments

In this subsection the estimates obtained are displayed. The ANN configuration used was the one that generated the smaller error Q, as presented in Tab. II, i.e. the ANN with 3 hidden layers, 100 neurons per hidden layer and ReLU as activation function. All the gap sizes are defined as 0.01 m - tests with gaps much smaller than that yielded poor results due to the lack of sensitivity. Furthermore, gaps that are too close are interpreted as only one, due to diffusion effects of the problem.

In Fig. 3 it is possible to see a good approximation of the gap position. Tests with other positioning and sizes yielded similar results to this one, indicating that the ANN approach have no problem in estimating one gap.

When two gaps are present, as displayed in 4, the estimates are also good in approximating the exact $\gamma(x)$ auxiliary function. But here the distance between the gaps have a minimum limit until they are considered as one. Again, gaps that are larger than the ones displayed are usually easier to estimate.

The estimate obtained for three gaps configuration is presented in Fig. 5. The estimate displayed also shows good approximation with the exact values, but it is possible to observe that this configuration generated an estimate for the two first gaps that are not really close to "0". Nevertheless, the positioning of all the three failures were reasonably recovered, which is the most important aspect of the technique.

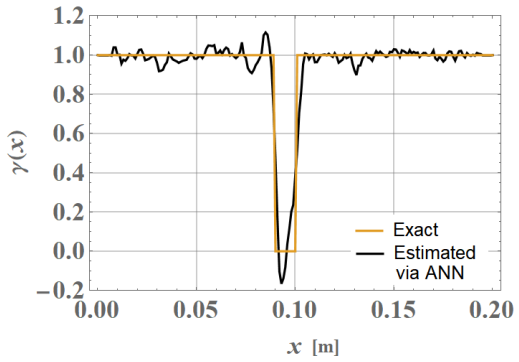


Fig. 3. Results obtained for the ReLU, 3 hidden layers, 100 neurons per hidden layer, and with a gap size of 0.01 m

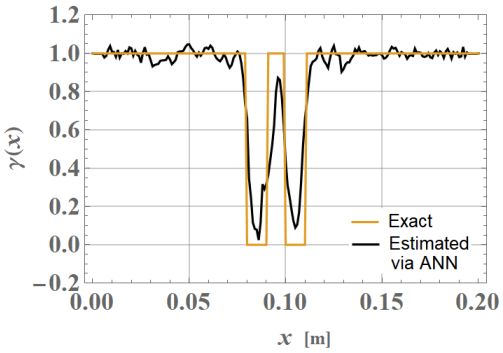


Fig. 4. Results obtained for the ReLU, 3 hidden layers, 100 neurons per hidden layer, and with two gaps with size of 0.01 m each.

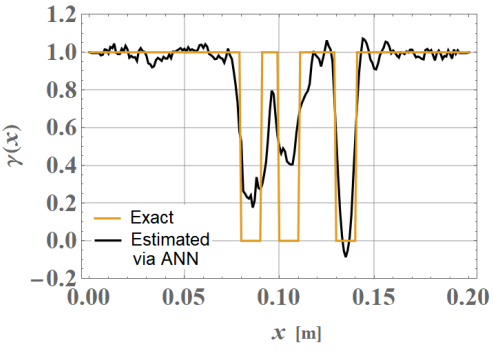


Fig. 5. Results obtained for the ReLU, 3 hidden layers, 100 neurons per hidden layer, and with three gaps with size of 0.01 m each.

Using the output of the ANN from Fig. 3 (case with 1 gap) and translating it back to the thermal properties $k(x, y)$ and $\rho(x, y)c_p(x, y)$ by rescaling to the exact thermal properties values of Tab. I, the direct problem is solved and the temperature profile obtained is displayed along the the simulated experimental data in Fig. 6. The maximum absolute errors between estimated and experimental temperature are 0.47 ºC

for $t = 1200$ s, 0.52 ºC for $t = 1500$ s, and 0.63 ºC for $t = 1800s$, which can be considered as a reasonable maximum errors for this temperature magnitude.

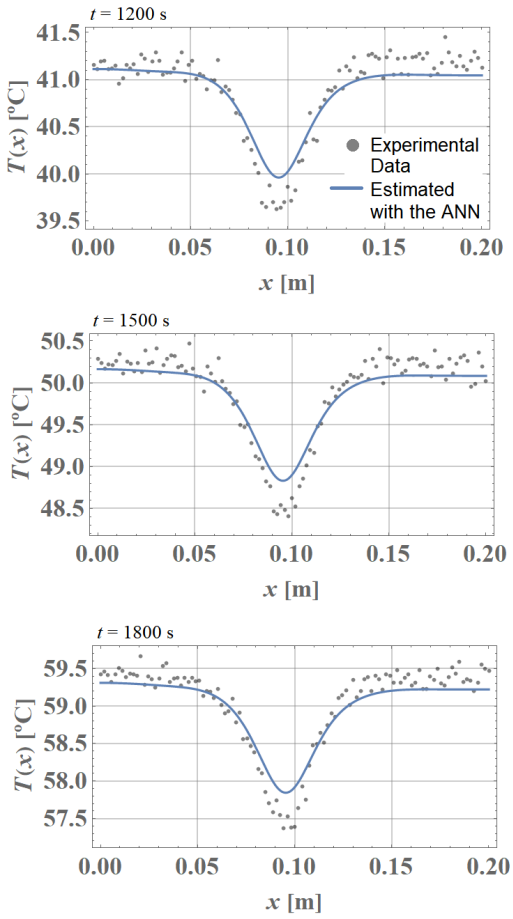


Fig. 6. Estimated temperature profile obtained with the ANN for the case with 1 gap and the respective simulated experimental data

V. Conclusions

The problem of identifying the position of gaps in multi-layered media was addressed using an Artificial Neural Network as inverse model. With this approach, the network was constructed with the hypothesis that the middle layer of the medium is either made of adhesive or air, meaning that all the thermal properties are known, but the positions of such gaps are not. With this assumption in mind, an auxiliary function that describes the position of the gaps was formulated and used as the Artificial Neural Network output, while the input were experimental temperature data of the exposed surface of the medium. Results have show that this is a feasible approach for identifying the positioning of several gaps, given that they are not close together. Moreover, the computational cost is mainly related to the generation of the training data and the training itself, so if the problem is defined and the network is properly trained, the estimation of the gap's positioning can be performed instantaneously, which is a desired feature for addressing the quality of adhesion in manufacturing facilities. Further investigations may focus on solving the same problem using a 3D formulation, where the experimental data is a two dimension thermographic image.

Acknowledgment

The authors acknowledge the financial support provided by CAPES - Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (Finance Code 001), CNPq - Conselho Nacional de Desenvolvimento Científico e Tecnológico, and FAPERJ - Fundação Carlos Chagas Filho de Amparo à Pesquisa do Estado do Rio de Janeiro. The authors acknowledge also the CAPES PrInt program grants (Process Code 88887.469279/2019-00 and 88887.311757/2018-00). This work has been also partially funded by the Spanish Ministry of Economy and Competitiveness with the support of the project TIN2017-86647-P (including funds from the European Regional Development Fund, ERDF).

References

[1] J. V. Beck, B. Blackwell and C. R. St. Clair, Inverse Heat Conduction: Ill-Posed Problems, New York, NY: Wiley Interscience, 1985.
[2] F. D. Moura Neto, and A. J. Silva Neto, An Introduction to Inverse Problems with Applications, Berlin, Springer-Verlag, 2013.
[3] J. P. Kaipio and C. Fox, "The Bayesian framework for inverse problems in heat transfer." Heat Transfer Engineering, vol. 32, no. 9, 2011, pp. 718–753.
[4] H. R. Orlande, "Inverse problems in heat transfer: New trends on solution methodologies and applications" Trans. ASME J. Heat Transf., vol. 134, no. 3, 2012, pp. 31011.
[5] H. R. Orlande, G. S. Dulikravich, M. Neumayer, D. Watzenig, and M. J. Colaço, "Accelerated Bayesian inference for the estimation of spatially varying heat flux in a heat conduction problem" Numer. Heat Transf. Part A: Appl., vol. 65, no. 1, 2014, pp. 1–25.
[6] S. Vakili and M. S. Gadala "Effectiveness and Efficiency of Particle Swarm Optimization Technique in Inverse Heat Conduction Analysis." Numerical Heat Transfer, Part B: Fundamentals, vol. 56, no. 2, 2009, pp. 119–141
[7] S. Garcia, J. Guynn and E. P. Scott. "Use of genetic algorithms in thermal property estimation: part II-simultaneous estimation of thermal properties." Numerical Heat Transfer, Part A Applications, vol. 33, no. 2, 1998, pp 149–168.
[8] A. Imani, A. A. Ranjbar and M. Esmkhani. "Simultaneous estimation of temperature-dependent thermal conductivity and heat capacity based on modified genetic algorithm." Inverse Problems in Science and Engineering, vol. 14, no. 7, pp. 767–783.
[9] F. J. C. P., Soeiro, P. O. Soares, H. F. Campos Velho and A. J. Silva Neto. "Using neural networks to obtain initial estimates for the solution of inverse heat transfer problems." In Inverse Problems, Design an Optimization Symposium, 2004, pp. 358–363.
[10] E. H. Shiguemori, J. D. S. Da Silva and H. F. Campos Velho, H. F. "Estimation of initial condition in heat conduction by neural network." Inverse Problems in Science and Engineering, vol. 12, no. 3, 2004, pp 317–328.
[11] C. Balaji, and T. Padhi. "A new ANN driven MCMC method for multi-parameter estimation in two-dimensional conduction with heat generation." International journal of heat and mass transfer, vol. 53, no. 23-24, 2010, pp. 5440–5455.
[12] B. Czél, K. A. Woodbury and G. Gróf. "Simultaneous estimation of temperature-dependent volumetric heat capacity and thermal conductivity functions via neural networks." International Journal of Heat and Mass Transfer, vol. 68, 2014, pp. 1–13.
[13] L. A. Abreu, H. R. Orlande, J. Kaipio, V. Kolehmainen, R. M. Cotta and J. N. Quaresma. "Identification of contact failures in multilayered composites with the Markov chain Monte Carlo method" Trans. ASME J. Heat Transf., vol. 136, no. 10, 2014, pp. 101–302.

[14] L. A. Abreu, M. J. Colaço, H. R. Orlande and C. J. Alves. “Thermography detection of contact failures in double layered materials using the reciprocity functional approach,” Appl. Therm. Eng., vol. 100, 2016, pp. 1173–1178.

[15] F. S. Mascouto, D. C. Knupp, L. A. Abreu, E. L. Sanches and L. A. Sphaier. “Detection of contact failures employing combination of integral transforms with single-domain formulation, finite differences, and Bayesian inference.” Numerical Heat Transfer, Part A: Applications, vol. 77, no. 6, 2020, pp. 599–618.

[16] D. C. Knupp, C. P. Naveira-Cotta, H. R. Orlande, and R. M. Cotta, “Experimental identification of thermo- physical properties in heterogeneous materials with integral transformation of temperature measurements from infrared thermography.” Exp. Heat Transf., vol. 26, no. 1, 2013, pp. 1–25.

Estimación de la biomasa en acuicultura utilizando redes neuronales convolucionales

Samuel López
Departamento de Tecnologías de la Información
Universidad de Huelva
Huelva – España
samuel.lopez@dti.uhu.es

Antonio Peregrín
Centro de Estudios Avanzados en Física, Matemáticas y Computación (CEAFMC)
Instituto Interuniversitario de Investigación en Data Science and Computational Intelligence (DaSCI)
Universidad de Huelva
Huelva – España
peregrin@uhu.es

Juan Carlos Gutiérrez-Estrada
Departamento de Ciencias Agroforestales
Universidad de Huelva
Huelva – España
juanc@dcaf.uhu.es

Inmaculada Pulido-Calvo
Departamento de Ciencias Agroforestales
Universidad de Huelva
Huelva – España
ipulido@dcaf.uhu.es

Jairo Castro-Gutiérrez
Departamento de Ciencias Agroforestales
Universidad de Huelva
Huelva – España
jairo.castro@dcaf.uhu.es

Fernando Gómez-Bravo
Departamento de Ingeniería Electrónica, Sistemas Informáticos y Automática
Universidad de Huelva
Huelva – España
fernando.gomez@diesia.uhu.es

Alejandro Garrocho-Cruz
Departamento de Ingeniería Electrónica, Sistemas Informáticos y Automática
Universidad de Huelva
Huelva – España
alejandro.garrocho@diesia.uhu.es

***Resumen*—La estimación de la biomasa y del crecimiento de los peces en entornos piscícolas son esenciales a la hora de preparar un plan de producción adecuado, permitiendo optimizar procesos tales como el cálculo de las tasas de alimentación o la descarga de nuevos lotes. Tradicionalmente esta tarea se ha hecho de forma manual, previo desembalse, suponiendo esto un gran coste tanto económico como humano. Se propone un método alternativo, automático y en tiempo real, mediante imágenes de sonar y usando técnicas de Deep Learning, para realizar dicha estimación sin necesidad de desembalse suponiendo esto un ahorro económico, así como una reducción de los tiempos empleados y un menor estrés para los animales.**

Palabras clave—Detección de objetos; Segmentación Semántica; Segmentación de Instancias; Convolutional Neural Networks; Deep Learning

I. Introducción

La sobrepesca, es decir, la captura en cantidades demasiado elevadas, además de un impacto directo sobre las especies objeto de captura tiene efectos devastadores sobre los ecosistemas como puede ser la destrucción de los arrecifes o la reducción de algunas especies. Como solución sostenible a este tipo de pesca surgió la acuicultura.

Podemos definir la acuicultura como el conjunto de técnicas aplicadas al cultivo de especies acuáticas. Este tipo de técnicas pueden ser llevadas a cabo en distinto entornos como puede ser agua dulce (ríos o lagos), agua salada o tanques de tierra.

Para desarrollar un plan de producción óptimo en este tipo de sistemas, es indispensable conocer la biomasa la cual se estima en su forma más simple multiplicando el peso medio de los peces por el número total de peces que existen en la balsa [1]. El cálculo de este índice es una operación muy costosa ya que en primer lugar se procede al desembalse, es decir, al vaciado total o parcial de la balsa, para luego mediante un muestreo manual establecer el peso medio de los peces.

Una posible solución a este problema consistiría en la automatización de este proceso haciendo uso de técnicas de *Deep Learning* las cuáles permiten obtener a través de transformaciones no lineales, representaciones óptimas de los datos con distintos niveles de abstracción para su posterior detección o clasificación [2]. Dentro de este campo, el *Deep Learning*, podríamos destacar las Convolutional Neural Networks (CNN en adelante) que son redes neuronales artificiales las cuáles fueron diseñadas para el procesamiento de datos 2D como por ejemplo imágenes digitales [2][3], aunque si se plantea en el dominio adecuado con esta red podría procesarse otros tipos de datos como puede ser el sonido.

En este trabajo se aplican técnicas de *Deep Learning* para el cálculo en tiempo real de la biomasa, en concreto haremos uso de una CNN e imágenes de sonar. Este tipo de sensores ofrece imágenes poco definidas y difíciles de interpretar, pero su uso viene justificado ya que en ambientes acuosos turbios no tiene cabida el uso de cámaras de visión.

Este trabajo está organizado de la siguiente manera: la sección *Antecedentes* repasa algunos conceptos claves empleados como por ejemplo el algoritmo *watershed*, en la sección *Modelo Propuesto* se presenta la metodología seguida para el cálculo de la biomasa, en la sección *Evaluación del Modelo* presentaremos los resultados obtenidos y las configuraciones usadas y finalizaremos con algunas conclusiones en la sección *Conclusiones*.

II. Antecedentes

En esta sección se va a repasar brevemente algunos de los conceptos empleados en nuestra propuesta como son la

segmentación semántica, la segmentación de instancias y el algoritmo *watershed*.

A. Segmentación semántica

Se conoce como segmentación semántica al conjunto de técnicas que tienen por objetivo asignar una etiqueta a cada uno de los píxeles que forman la imagen, de tal modo que dos píxeles que corresponden a objetos de la misma categoría tendrán asignada la misma etiqueta.

Cabe destacar que este tipo de técnicas permiten conocer la clase a la que pertenece un píxel, pero no permite separar instancias dentro de la misma clase. En otras palabras, si en una imagen tenemos varios objetos del mismo tipo, a la salida obtendremos un mapa de segmentación en el cual todos los píxeles correspondientes a dichos objetos tendrán la misma etiqueta.

Este tipo de técnicas son ampliamente usadas en entornos médicos como puede ser para la detección de tumores [4] o para la segmentación de células [5].

B. Segmentación de instancias

La segmentación de instancias a diferencia de la segmentación semántica permite etiquetar los píxeles que forman una imagen según su categoría, pero además para objetos de la misma categoría permite distinguir las distintas instancias.

Según su enfoque, estas técnicas pueden ser agrupadas en los siguientes subgrupos [6]:

- *Generación de máscaras seguida de clasificación*. El modo de funcionamiento de estas técnicas se basa en la generación de un conjunto de máscaras, una por cada objeto que forme parte de la imagen, para luego pasarlas a un clasificador y saber de qué objeto se trata. Dentro de este grupo encontramos arquitecturas como R-CNN [7], Fast R-CNN [8] y Faster R-CNN [9].
- *Detección seguida de segmentación*. Los algoritmos encuadrados dentro de este grupo basan su funcionamiento en generar un conjunto de cajas con las coordenadas de los objetos para seguidamente realizar la segmentación de este. Un ejemplo de este tipo de redes lo encontramos en Mask R-CNN [10].
- *Etiquetado de pixeles seguido de clusterizado*. Este grupo de técnicas se apoyan en las técnicas utilizadas para segmentación semántica. En primer lugar, se realiza una clasificación de todos los píxeles que aparecen en la imagen indicando su categoría. Acto seguido, y utilizando técnicas de *clusterizado,* se agrupan los píxeles adyacentes con las mismas etiquetas formando las distintas instancias. Un ejemplo de este grupo lo encontramos en [11] y [12].
- *Métodos basados en el desplazamiento de ventanas*. Este conjunto de técnicas basa su funcionamiento en desplazar una serie de ventanas con distintas escalas y relaciones sobre la imagen generando un conjunto de máscaras, usando para ello una CNN. Posteriormente estas máscaras son pasadas a un clasificador. [13] y [14] son ejemplos de este grupo.

C. Algoritmo watershed

El algoritmo *watershed* [15][16] es un método de segmentación orientada a regiones que permite extraer las fronteras de las regiones existentes en una imagen.

Este algoritmo trata de buscar hacia que mínimo local o valle iría una gota que cae sobre un relieve topográfico. De forma análoga, una imagen puede ser vista como un relieve topográfico donde la intensidad de cada píxel representa su altura.

El modo de funcionamiento consiste en partir de un conjunto de nodos semillas, los cuales corresponden a mínimos locales, a los que se les va añadiendo los píxeles vecinos que presentan características similares.

III. Modelo Propuesto

La estimación de la biomasa de un estanque se realiza a través de una evaluación de la cantidad y peso de los peces que se observan a través de un sonar de imagen, y es en este paso de procesamiento de la información captada por el sonar donde empleamos un método basado en segmentación de instancias a través del etiquetado de pixeles utilizando una CNN. En particular, se realiza una segmentación semántica seguido de técnicas de clusterizado para separar los diferentes peces. En las siguientes subsecciones se detalla la arquitectura empleada y las particularidades de este proceso.

A. Arquitectura

Los *autoencoders* son un tipo de red neuronal que fue introducida por primera vez en [17] y que tiene por objetivo aprender de una forma no supervisada como comprimir, de una forma óptima, los datos de entrada en un espacio latente, y a continuación, aprende a reconstruir los datos de salida a partir de esta información.

Este tipo de red se dividen en dos partes tal y como aparece en la Fig. 1:

- *Encoder*. Parte encargada de comprimir la entrada.
- *Decoder*. Parte encargada de reconstruir la salida en función de la información previa.

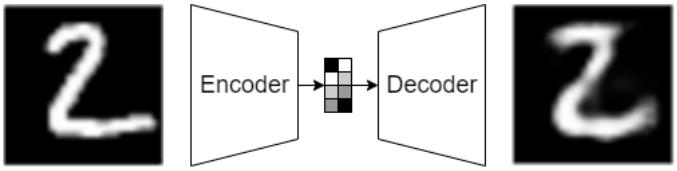


Fig. 1. Arquitectura de un *autoencoder*. Los datos de entrada son comprimidos por un encoder para posteriormente reconstruir la salida

Una U-Net [5] es un tipo de *autoencoder* usado principalmente para tareas de segmentación semántica y que cuenta con la particularidad de tener conexiones desde las primeras capas hacia capas más avanzadas permitiendo la propagación de la información.

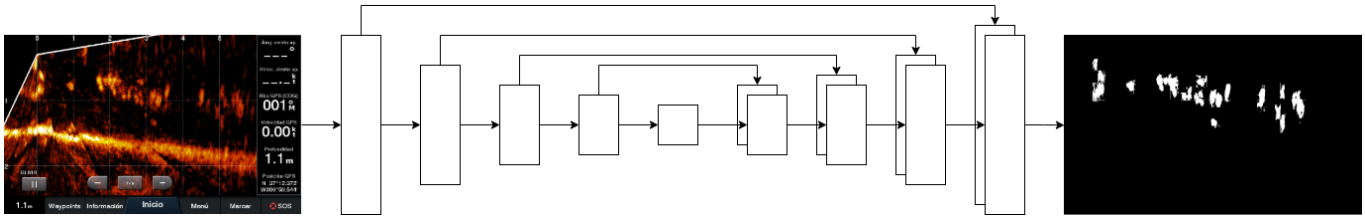


Fig. 2. Arquitectura propuesta. La red está formada por dos partes, la primera es la encargada de comprimir la imagen reduciendo su dimensionalidad generando un mapa de características, mientras que la segunda, es la encargada de reconstruir la imagen de salida a partir de la información previa.

La arquitectura empleada en esta aplicación ha sido la que aparecen la Fig. 2 la cuál toma como base una U-Net. Esta red es usada para separar los píxeles correspondientes a pez de los píxeles correspondientes a fondo.

Una vez segmentados los píxeles correspondientes a pez de los píxeles correspondientes a fondo, se ha empleado el algoritmo *watershed* con la idea de identificar los distintos especímenes que aparecen en la imagen. Un ejemplo de los resultados obtenidos en este proceso puede ser visto en la Fig. 3.

B. *Cálculo de la biomasa*

En el cálculo de la biomasa es indispensable conocer el peso de los peces, así como el número de individuos con tamaño comercial.

En entornos con una única especie, como es el caso de este trabajo, existe una relación directa entre la altura del pez y su peso. Para hallar la altura de cada pez solo sería necesario recorrer las distintas etiquetas generadas y por cada una hallar su altura en píxeles. A continuación, esta medida es convertida en altura real conociendo la relación de escala proporcionada por el sonar. A partir de esta transformación es fácil calcular la longitud real del pez sabiendo que:

$$L_T = 2.7711 * H + 2.3891, \quad (n = 57;\ r^2 = 0.9387;\ p < 0.05) \qquad (1)$$

$$W = 0.054 * L_T^{3.31}, \quad (n = 57;\ r^2 = 0.9787;\ p < 0.05) \qquad (2)$$

donde L_T es la longitud total (cm); H es la altura total entre la aleta dorsal y el comienzo de las aletas pélvicas (cm) y W es el peso fresco (g). Estas relaciones fueron obtenidas experimentalmente a partir de ejemplares capturados en la piscifactoría en estudio.

Los sonar de imagen son dispositivos que permiten medir profundidades y las distancias hasta los objetos que se encuentren bajo el agua. Estos tipos de sonar realizan un escaneo sobre un espacio 3D. El volumen de este espacio puede ser calculado conociendo el ángulo de apertura del dispositivo. Esta característica ha sido el punto de partida para el cálculo del número total de ejemplares.

Para hallar el total de peces en el estanque, se ha calculado en primer lugar la densidad de especímenes por metro cúbico en la región cubierta por el haz del sonar. Esto se consigue tomando imágenes durante un periodo de tiempo y calculando la media de peces que aparecen en la escena. Esta densidad es calculada mediante la función:

$$\overline{D_t} = 1.66 \frac{360 n_t}{\pi r^2 \alpha h} \qquad (3)$$

donde $\overline{Dt}$ es la densidad estimada en el momento t; r es el radio de detección del sonar; α es el ángulo que determina la anchura del haz del sonar; y h es la altura media de la balsa. Se aplica un factor de corrección de 1.66 debido al concepto de 'cono de detectabilidad', el cual establece que solo el 66% de los especímenes son detectados. A continuación, sabiendo el volumen de la balsa, extrapola el total de especímenes multiplicando el valor de esta densidad por el volumen total del estanque.

Con las imágenes recogidas en ese mismo periodo, se calculan los pesos de los peces de la forma descrita y se halla una distribución de los distintos pesos. Con esta distribución y el total de especímenes se calcula la biomasa.

IV. Evaluación del Modelo

En esta sección se describen las pruebas llevadas a cabo para comprobar el funcionamiento del modelo propuesto.

A. *Configuración de las pruebas*

En la fase de aprendizaje de nuestro modelo hemos utilizado un servidor con dos procesadores Xeon de 20 núcleos cada uno,



Fig. 3. Resultados obtenidos después de aplicar el algoritmo *Watershed*. De izquierda a derecha: imagen de entrada, fuente de verdad y predicción.

512GB de RAM y 10 GPUs NVIDIA RTX QUADRO 6000 con 24GB, aunque el aprendizaje se hecho de forma local, es decir, un modelo por GPU.

El conjunto de datos empleado está formado por 4039 imágenes de sonar. La recogida de estas imágenes se ha realizado de forma estática, es decir, se ha sumergido el sensor y siempre desde la misma posición se han tomado las distintas imágenes (Fig. 4).



Fig. 4. Transductor del sonar de imagen LVS32 de Garmin, acoplado a un módulo LiveScope GLS10 (Sistema Panoptix LiveScope™).

Aunque se dispone de una cantidad suficiente de imágenes, se han usado técnicas de *data augmentation* ya que en las imágenes del sonar aparece el fondo de la balsa, el cual tiene una pendiente. Para evitar el aprendizaje de esta zona por parte de la red neuronal, se ha hecho una rotación con respecto al eje vertical del 50% de las imágenes de tal forma que la pendiente aparece en ambos lados.

El proceso de generación de las máscaras de salida ha consistido en primer lugar en la generación de forma manual de un conjunto de cajas, conteniendo cada una de ellas un único espécimen. Aplicando el algoritmo *k-means*, se ha dividido la imagen en 6 clústeres; la elección del número de clústeres se ha realizado siguiendo el método de *elbow* [18]. A continuación, se han seleccionado los píxeles que pertenecen a los clústeres que son de nuestro interés y que se encuentran dentro de las cajas que se han generado previamente.

B. *Métricas para validar el modelo*

Inspirados en [19], para la evaluación de nuestro modelo se hace uso de diferentes métricas, concretamente: *Dice Coefficient*, *Sensitivity* y *Specificity*.

- *Dice Coefficient (DC).* Mide el nivel de solapamiento entre el mapa de segmentación predicho y la fuente de verdad. Es definida por la función:

$$DC = \frac{2TP}{2TP + FP + FN} \tag{3}$$

- *Sensitivity (TPR).* Calcula el índice de las predicciones positivas correctas entre el total de positivos. Es definida por la función:

$$Sensitivity(TPR) = \frac{TP}{TP + FN} \tag{4}$$

- *Specificity (TNR).* Calcula el índice de las predicciones negativas correctas entre el total de negativos. Es definida por la función:

$$Specificity(TNR) = \frac{TN}{TN + FP} \tag{5}$$

C. *Ajuste del modelo*

Para el ajuste de nuestro modelo, se ha hecho uso del optimizador Adam con los siguientes parámetros: $\alpha = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$ y $\epsilon = 10^{-8}$; el artículo que propuso esta técnica [20] fijaba estos valores, y son los que hemos usado. Así mismo, se ha hecho uso de tres funciones de coste con idea de buscar cuál es la que mejor se ajusta a nuestras imágenes. En la elección de estas funciones de coste han influido las características de nuestro problema: clasificación binaria con un gran desbalanceo entre las clases. Las funciones de coste empleadas son:

- *Binary Cross-Entropy.* La entropía cruzada [20] se define como la diferencia entre dos distribuciones de probabilidad para una variable aleatoria. Es definida por la función:

$$L_{BCE}(y, \hat{y}) = (y \log(\hat{y}) + (1 - y) \log(1 - \hat{y})) \tag{6}$$

- *Dice Loss* [21]. Esta función de coste es una adaptación de *Dice Coefficient*. Es definida por la función:

$$DL(y, \hat{p}) = 1 - \frac{2y\hat{p} + 1}{y + \hat{p} + 1} \tag{7}$$

- *Focal Loss* [22]. Variante de *Binary Cross-Entropy*. Es utilizada en entornos con desbalanceo de clases. Es definida por la función:

$$FL(p_t) = -\alpha_t (1 - p_t)^{\gamma} \log(p_t) \tag{8}$$

D. *Resultados obtenidos y análisis*

La Tabla I muestra los resultados obtenidos para las tres funciones de coste para las tres métricas propuestas.

TABLA I. COMPARACIÓN DE LAS FUNCIONES DE COSTE USADAS EN FUNCION DE LAS DISTINTAS MÉTRICAS DE EVALUACIÓN

Función de coste	**Métricas de evaluación**		
	DC	***TPR***	***TNR***
Binary Cross-Entropy	0.8669	0.7850	0.9995
Dice Loss	0.7778	0.7442	0.9968
Focal Loss	0.8702	0.7888	0.9995

Se observa que los mejores resultados se obtienen para la función de coste *FL*. Esto es debido principalmente a que esta función fue diseñada para tratar con problemas en los que existe un gran desbalanceo entre las clases como es el caso que nos

ocupa, habiendo muchos más píxeles correspondientes a fondo que píxeles correspondientes a pez.

Por el contrario, el peor resultado se obtiene para la función de coste *DL*. Esta función es una adaptación del *DC*, el cual se basa en el nivel de solapamiento de la predicción con la fuente de verdad. Este valor tan bajo con respecto a las otras dos funciones de coste puede deberse al hecho de que no existan unos bordes definidos y es aquí donde se comete el error.

Observando las imágenes de salida se ven segmentos identificados como peces que inicialmente no estaban etiquetados. Esto se observa principalmente en algunas imágenes con secciones de ruido relativamente grandes. No lo consideramos relevante ya que a ojos de un experto, es difícil discernir si realmente es ruido o si se trata de un pez.

En cuanto a la capacidad del modelo para predecir la biomasa en el estanque en el que se ha probado, el resultado estimado es de 11141 peces con un peso promedio de 238 g y una biomasa total de 2.6 tn. Estos resultados son muy próximos a los valores suministrados y estimados por los gestores de la piscifactoría (unos 15000 peces de 250 g). Hay que tener en cuenta que es muy difícil conocer la cantidad real de peces introducidos en un estanque ya que ésta es una aproximación realizada por los gestores a partir de un peso total del lote introducido y un peso promedio. Este procedimiento por regla general sobrestima el número de ejemplares en el lote. Así mismo, hay una cantidad significativa de pérdidas no contabilizadas como consecuencia de la mortalidad natural y la causada por factores como el efecto de enfermedades, depredación por aves piscívoras y hurtos. Por tanto, se puede asumir que los resultados se aproximan significativamente al número de ejemplares presentes en el estanque.

V. Conclusiones

El objetivo de este trabajo consiste en establecer un método con el que estimar la biomasa en estanques con baja visibilidad en piscifactorías empleando un sensor de tipo sonar de imagen. Se propone llevarlo a cabo procesando las imágenes de baja resolución mediante técnicas de *Deep Learning* para obtener datos que permiten calcular el peso de los peces y su cantidad.

El resultado de este estudio es positivo, es decir, se ha conseguido segmentar los peces que aparecen en las imágenes del sonar. Dado que el modelo propuesto consigue un gran nivel de solapamiento de las imágenes de salida con las imágenes objetivo, es válido para estimar los pesos de los peces de una forma fidedigna.

Queda pues pendiente completar este estudio contrastando la estimación que obtiene el modelo con la biomasa real en los estanques, de modo que sea cuantificable la calidad de la medida realizada por este procedimiento más allá de la primera prueba realizada presentada en este documento. Por otro lado, también es posible extender este trabajo al uso de imágenes de vídeo en lugar de solamente estáticas, ya que consideramos que gran parte de la información para segmentar un pez en una imagen en estas condiciones, está también relacionada con el movimiento de los propios animales.

Agradecimientos

Este trabajo se corresponde con una parte de los resultados del Proyecto KTTSeaDrones cofinanciado por el Fondo Europeo de Desarrollo Regional FEDER a través del programa Interreg V-A España-Portugal (POCTEP) 2014-2020, con ref. 0622-KTTSEADRONES-5-E.

Asimismo, este trabajo ha sido financiado por el Ministerio de Ciencia e Innovación en el marco del proyecto de investigación con ref. TIN2017-89517-P.

Referencias

[1] M. Haddon, “Modelling and quantitative methods in fisheries”. Second edition, Ed. Taylor and Francis Group, LLC, Boca Raton, 449 pp, 2011.

[2] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning”, Nature, vol. 521, no. 7553, p. 436, 2015.

[3] Y. Lecun, L. Bottou, Y. Bengio and P. Haffner, “Gradient-based learning applied to document recognition,” in Proceedings of the IEEE, vol. 86, no. 11, pp. 2278-2324, 1998.

[4] M. Havaei, A. Davy, D. Warde-Farley, A. Biard, A. Courville, Y. Bengio, and H. Larochelle, “Brain tumor segmentation with deep neural networks”. Medical Image Analysis, 35, 18-31, 2017.

[5] O. Ronneberger, P. Fischer and T. Brox, “U-net: Convolutional networks for biomedical image segmentation”, in International Conference on Medical Image Computing and Computer-Assisted Intervention, pp. 234-241, 2015.

[6] A. M. Hafiz And G. M. Bhat, “A survey on instance segmentation: state of the art”. International Journal of Multimedia Information Retrieval, 1-19, 2020.

[7] R. Girshick, J. Donahue, T. Darrell and J. Malik, “Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation”, in IEEE Conference on Computer Vision and Pattern Recognition, pp. 580-587, 2014.

[8] R. Girshick, “Fast R-CNN”, in IEEE International Conference on Computer Vision (ICCV), pp. 1440-1448, 2015.

[9] S. Ren, K. He, R. Girshick and J. Sun, “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks”, in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 6, pp. 1137-1149, 2017.

[10] K. He, G. Gkioxari, P. Dollár and R. Girshick, “Mask R-CNN”, in IEEE International Conference on Computer Vision (ICCV), pp. 2980-2988, 2017.

[11] M. Bai and R. Urtasun, “Deep Watershed Transform for Instance Segmentation”, in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2858-2866, 2017.

[12] A. Kirillov, E. Levinkov, B. Andres, B. Savchynskyy and C. Rother, “InstanceCut: From Edges to Instances with MultiCut”, in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 7322-7331, 2017.

[13] P. O. Pinheiro, R. Collobert and Piotr Dollár, “Learning to segment object candidates”, in Proceedings of the 28th International Conference on Neural Information Processing Systems, vol. 2, pp. 1990–1998, 2015.

[14] J. Dai, K. He, Y. Li, S. Ren and J. Sun, "Instance-sensitive fully convolutional networks", in European Conference on Computer Vision, pp. 534-549, 2016.

[15] L. Vincent and P. Soille, “Watersheds in digital spaces: an efficient algorithm based on immersion simulations”, in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 13, no. 6, pp. 583-598, 1991.

[16] R. Romero-Zaliz and J. F. Reinoso-Gordo, “An updated review on watershed algorithms” in Soft computing for sustainability science, pp. 235-258, 2018.

[17] D. E. Rumelhart, G. E. Hinton and R. J. Williams, “Learning internal representations by error propagation”. California Univ San Diego La Jolla Inst for Cognitive Science, 1985.

[18] R. L. Thorndike, “Who belongs in the family?,” Psychometrika, vol.18, no.4, pp.267–276, 1953.

[19] S. Jadon, “A survey of loss functions for semantic segmentation”, in IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB), pp. 1-7, 2020.

[20] D. P. Kingma and L. Ba Jimmy, “Adam: A method for stochastic optimization”, 2014. arXiv preprint arXiv:1412.6980. M. Yi-de, L. Qing and Q. Zhi-bai, “Automated image segmentation using improved PCNN model based on cross-entropy”, Proceedings of 2004 International Symposium on Intelligent Multimedia, Video and Speech Processing, pp. 743-746, 2004.

[21] C. H. Sudre, W. Li, T. Vercauteren, S. Ourselin and M. J. Cardoso, “Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations”, In Deep learning in medical image analysis and multimodal learning for clinical decision support (pp. 240-248), 2017.

[22] T. Lin, P. Goyal, R. Girshick, K. He and P. Dollár, “Focal Loss for Dense Object Detection”, in IEEE International Conference on Computer Vision (ICCV), pp. 2999-3007, 2017.

An unsupervised anomaly detection strategy for industrial processes under data shift scenarios

1st Xabier Barrenetxea Berasategi
TECNALIA (BRTA)
Basque Research & Technology Alliance
20009 San Sebastian, Spain
xabier.barrenetxea@tecnalia.com

2nd Alberto Diez-Olivan
TECNALIA (BRTA)
Basque Research & Technology Alliance
20009 San Sebastian, Spain
alberto.diez@tecnalia.com

3rd Óscar Cordón
DECSAI and
Andalusian Research Institute DaSCI
University of Granada
18071 Granada, Spain
ocordon@decsai.ugr.es

***Abstract*—Anomaly detection and prediction is a key factor for reducing nonconformity costs in industrial processes. In the manufacturing sector multiple manufacturing references can be produced, each of them comprising specific and unique configuration setups that may imply heavy changes in the production process. Therefore, the same process fluctuates into different manufacturing setting points that modify the distribution of the data, which is known as data shift. The problem becomes even more difficult if there is no previous information about the kinds of anomalies that can arise in the manufacturing process. Data shift scenarios make it challenging to deploy an anomaly detection model which can work efficiently over time, thus requiring adaptive strategies to overcome the problem. This paper presents an adaptive learning strategy for unsupervised anomaly detection based on the use of an autoencoder. The autoencoder allows us to grasp all the nonlinearities related to the process data, while the adaptive learning strategy generalizes the model to fit many different manufacturing shifts optimally. Experiments carried out on a real manufacturing problem showed that the use of the adaptive learning strategy reduces the model error and improves the accuracy of the static model.**

***Index Terms*—Manufacturing, anomaly detection, adaptive learning, data shift, autoencoders, kernel density estimation.**

I. Introduction

Manufacturing and production processes are complex and very sensitive to changing environmental conditions and to the different parameters that define them. In many cases, for instance, various references are manufactured in the same production line and therefore, the different variables intrinsic to the process and their distribution do not remain constant. This is known as data shift. The main reasons that originates data shift are bound to changes in the physical properties of the process (material, dimensions, quantity, etc.), adjustments of machine set-up or due to replacement of commercial tools used from different suppliers. To solve this problem requires self-adaptive intelligent models that can quickly learn the new environment to predict events of interest that affect the quality of the industrial processes' outputs.

The said problem makes it necessary to have algorithms that can adapt to changing and unknown situations in complex industrial manufacturing and production environments [1]. The training of these models must be executed automatically and in reasonable short time. Therefore, the choice of the intelligent models to be applied and the training data to fit them is crucial in order to allow the model to adapt to new situations without neglecting its historical behavior. Besides, the designed solution must consider the fact that in many cases there is not *a priori* information about the anomalies that can arise in these manufacturing and production processes. The solution adopted consists of an artificial intelligence-based approach that takes into consideration all those constraints, by applying online learning methods able to fast and accurately predict quality nonconformities beforehand [2].

The autoencoders have the potential of extracting key information in the latent space that can be either linear or nonlinear within the multivariate data [3]. The proposed unsupervised anomaly detection strategy exploits the potential of autoencoders and computes the reconstruction error to setup a threshold for the anomaly detection. Furthermore, through adaptive learning strategies [4], the proposed algorithm is capable to quickly self-adapt to data shifts that occurs in industrial manufacturing scenarios. After a careful design process, the autoencoder defined for the case study considered in this paper is built by five fully connected dense layers with ReLU activation functions for the internal layers and the latent layer is composed of three neurons. The offline training of the autoencoder is done in an unsupervised fashion using a training dataset that includes four data shifts, which are not considered by the autoencoder. Consequently, all four shifts are considered to be drawn from the same distribution for training purposes. This pre-trained model is cached and it is used as the base input model within the adaptive learning strategy. Each time a data shift occurs, a small sample for a given window size of the newly distributed data is buffered to feed the adaptive learning strategy. This buffered data is fitted into a Kernel Density Estimator (KDE) to generate synthetic augmented data [5], which is later used in the adaptive process of the autoencoder. The adapted layers of the autoencoder are the latent layer and the decoding layers, that are refitted with a reduced number of epochs, thus making the adaption smooth and fine grained.

To evaluate the presented strategy an experiment on real industrial process has been carried out. The industrial process belongs to a water jet cutting machine, that is used to manufac-

This project was supported by the Basque Government through EMAITEK and ELKARTEK (ref. KK-2020/00049) funding grants.

ture plastic components for the automotive sector. The results of the experiments have shown that the adoption of the strategy has major benefits.

The paper is organized as follows. Section II reviews some state-of-the-art research in anomaly detection, autoencoders, and adaptive learning strategies. Section III introduces the proposed novel approach and its mathematical modelling. Section IV presents the experimental setup. Section V discusses the experimental results. Finally in section VI the conclusions and suggestions for future research are made.

II. Related Work

Anomaly detection and prediction is one of the main core areas of the fourth industrial revolution, and much research is being done related to the topic [6]. Manufacturing anomalies are always a drawback and they cost not only money, but also an increment in waste generation, delays on deliveries, loss of confidence from the customer, and frustration within the company [7]. Hence, the future seems to tend towards zero defect manufacturing [8], requiring shop floor production plans to be executed with the support of intelligent systems, making available key production information that is only accessible through digitization and artificial intelligence algorithms. In addition to the economic benefits, the adoption of that key enabling technologies will also imply a more sustainable production, which is becoming a primary driver during the current decade [9].

In recent years the irruption of a new wave on artificial neural networks has pushed beyond the limits of the research and development in the field [10]. This has led to new variants and applications of networks that have established a completely new framework of possibilities. Perhaps, one of the most promising network models are autoencoders, which seek to learn a compressed representation of an input, reducing its dimensionality *(Encoding)*, and reconstruct back the original dimension *(Decoding)*, by self-supervising. Autoencoders where first introduced in [11] and are used in many applications, such as dimensionality reduction [12], information retrieval [13], anomaly detection [14], image compression [15], and machine translation [16], among many others. The applicability of autoencoders in the industrial anomaly detection problem is broad and a significant amount of research has been done in the field, either in a supervised [17] or unsupervised [18] fashion.

Meanwhile, the digitization of factories has increased rapidly in the last decade, making available a large amount of historical data. In general all this data is suitable to generate predictive machine learning models, but they tend to predict from what they have seen in the past. Therefore, the resulting models are very dependent on the historical data, and they are unable to properly generalize when novel, unseen data arrives. This phenomenon is known as data shift. Data shift occurs when there is a change on the data distribution between the training and test data, resulting into degradation of the model and poor outcomes [19]. These distribution changes can occur due to different reasons and are not intrinsic to industrial environments and extend to any field where machine learning models are applied. Hence, more research is undergoing to vanquish the models obsolescence [20] [21].

It can thus be acknowledged that huge efforts have been done in research on industrial anomaly detection and anticipation, development of new neural network models and extension of their applications, as well as to give response to the data shift issue. Our approach, which involves an unsupervised anomaly detection strategy under data shift scenarios based on autoencoders, encompasses all the three research lines into one. After an extensive effort to find research within industrial environments with data shift, that merges the power of auntoencoders and adaptive learning into a single algorithm, bare information have been found, confirming that more research in the field is needed.

III. Adaptive Learning Strategy for Anomaly Detection

The proposed strategy contains the following key components:

- *A multilayer perceptron autoencoder optimally trained.* The autoencoder (AE) is trained offline with historical data. The given model is saved and it is adapted to every new shift by applying the adaptive learning strategy.
- *An adaptive learning strategy implementation.* For the adaptive learning strategy a set of data that represents the changed distribution is required. This data is generated synthetically from a small subset of the original data belonging to the new shift, that is buffered at the beginning for a specified window size. The technique used for synthetic data generation is based on a KDE. The generated synthetic data is then used to refit the latent and the decoder layers of the decoder.
- *Anomaly detection methodology.* Anomaly detection is done based on statistical methods, as we are working on unsupervised data. First, from the training data, three thresholds are established for the 0.95, 0.99, and 0.999 quantiles of the reconstruction error, enabling to label the anomalies in a simple three-tier system: minor, major, and critical. Finally, during the testing, the computed reconstruction error is compared to the thresholds.

The components mentioned above are fully described in Algorithm 1.

A. Autoencoder Architecture and Hyper-parameters

The chosen model for the key pattern extraction and to get a measurable metric from those patterns is a multilayer perceptron autoencoder, as it suits perfectly within the environment of industrial shop floor data. First, as a key property of artificial neural networks, it is able to extract non linear relationships within multivariate data. This provides a huge advantage against other methods, that only consider linear inter variable relationships, such as principal component analysis (PCA). On the other hand, artificial neural networks represent an ideal framework to implement adaptive learning strategies as they allow us to fine tune many hyper-parameters within the transfer

Algorithm 1 Adaptive AE-based anomaly detection

Input 1: shift information $C = \{C_1, ..., C_k\}$
Input 2: historical data $X = \{X_1, ..., X_m\}$
Input 3: streaming data $S = \{S_1, ..., S_k\}$
Input 4: window size W
1: **Step1** *(Offline AE model fit)*
2: Set the AE architecture $\rightarrow AE$
3: Standarize historical data $Z = X.Standarize()$
4: Off-line fitting of the AE on historical Z
5: Set T from $MSE_X \rightarrow T = \{T_{0.95}, T_{0.99}, T_{0.999}\}$
6: Make latent layer and decoder layers trainable
7: **Step2** *(Active Learning Anomaly Detection)*
8: **for** S_i in process data stream $S = \{S_1, ..., S_k\}$ **do**
9: **if** $C_{i-1} \neq C_i$ **then**
10: $refitSample \rightarrow []$
11: $refitSample.push(S_i)$
12: **else if** $i < W$ **then**
13: $refitSample.push(S_i)$
14: **else if** $i = W$ **then**
15: $refitSample.push(S_i)$
16: Generate synthetic dataset Q using Equation 1
17: Standarize synthetic data $Q.Standarize()$
18: Fit L_{Latent} and $L_{Decoder} \rightarrow AE'$
19: **else if** $i > W$ **then**
20: $S' = AE'(S.Standarize()).Standarize^{-1}()$
21: Reconstruction error $\rightarrow MSE_i = \sum_1^{17}(s_j - s'_j)$
22: **if** $MSE_i > T$ **then**
23: **return** Anomalies $\rightarrow \{A_{0.95}, A_{0.99}, A_{0.999}\}$
24: **end if**
25: **end if**
26: **end for**

learning process, resulting in a controlled adaptation. From all the autoencoder variants: multilayer perceptron (MLP) autoencoder, variational autoencoder [22], convolutional autoencoder [23], and convolutional variational autoencoder [24], the chosen autoencoder is the MLP autoecoder. This model is less complex than the others but perfectly fits the problem that we are trying to solve, where multivariate data is coming from different Industrial Internet of Things (IIoT) sensors. The optimal hyper-parameters of the model have been set on the basis of an exhaustive grid search and are dependent on the problem.

The proposed autoencoder is composed of five layers, two layers in the encoder and the decoder, and one in the latent space layer. The encoder and the decoder have 17 neurons in the input and output layers respectively (corresponding to the number of variables of the considered industrial process) and the latent layer is comprised by three neurons, as it can be seen in autoencoder architecture schema in figure 1. The ReLU activation functions are used in the internal layers of the autoencoder and the input and output layers have no activation functions. During the offline training phase, the learning rate α is set to 0.001, the number of epochs used is 100, and the early stopping regularization is implemented. The chosen loss function is the Mean Squared Error (MSE). All the hyper-parameters for the offline training are shown in Table I.

TABLE I
AE TRAINING HYPER-PARAMETERS

Hyperparameter	Value
Input layer Nº of neurons	17
Output layer Nº of neurons	17
Encoder internal layer Nº of neurons	8
Decoder internal layer Nº of neurons	8
Latent layer Nº of neurons	3
Input layer activation function	None
Output layer activation function	None
Encoder internal layer activation function	ReLU
Decoder internal layer activation function	ReLU
Latent layer activation function	ReLU
Optimizer	Adam
Loss function	Mean Squared Error
Regularization	Early Stopping: patience 5
Encoder layer trainable	True
Decoder layer trainable	True
Latent layer trainable	True
Batch size	32
Number of epochs	100

The training data is standardized using the formula $Z = (X - \mu_x)/std_x$. The standardized data Z_i, Line 3 in Algorithm 1, speeds up the convergence and optimizes the training phase of the network. The same standardization is applied during the test phase, Lines 16 and 19 in Algorithm 1.

B. Adaptive learning strategy

The adaptive learning strategy relies on a key parameter, the window size W, Input 4 to Algorithm 1. The window size sets the amount of original data that will be used during adaptation to the new shift, whenever a data shift happens, Line 9 in Algorithm 1. The shift information, Input 1 to Algorithm 1, is given beforehand. Once the streaming data, Input 3 to Algorithm 1, reaches the window size, Line 14 in Algorithm 1, the $refitSample$ is complete, Line 15 in Algorithm 1. Since the window size W aims to be small, so that the adaptation to the new shift occurs as quickly as possible, a synthetic dataset Q, Line 16, based on $refiSample$ data is generated, using a KDE. The KDE needs to determine a bandwidth, which has a direct impact on the resulting estimate of the KDE, and therefore on the generation of the synthetic data, which will be used to fit the adapted AE', Line 18 in Algorithm 1. The selection of the window size and the bandwidth is specific and unique for each industrial process. For every new shift 100 samples are generated with the KDE. The used Gaussian KDE is specified in Equation 1:

$$\hat{f}(\mathbf{x}; \mathbf{H}) = \frac{1}{n} \sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{x} - \mathbf{X}_i) \quad (1)$$

where the Gaussian kernel $K(\mathbf{z}) = \phi(\mathbf{z}) = (2\pi)^{-p/2} e^{-\frac{1}{2}\mathbf{z}'\mathbf{z}}$; $\mathbf{H}$ is the bandwidth, that behaves similar to the covariance matrix; $\mathbf{X}_1, \ldots, \mathbf{X}_n$ is the multivariate data, which in our case is the $refitSample$, Line 15 in Algorithm 1; and x is the data sample of which the probability estimate is returned.

The synthetic data Q, generated with the gaussian KDE, is standardized before the transfer learning process, Line 17 in Algorithm 1. The standardization of the data is done by using the equation introduced in this section, and μ_x and the std_x are respectively the median and the standard deviation calculated from the training dataset $X = \{X_1, ..., X_m\}$.

The final step of the adaptive learning strategy is the knowledge transfer from the pretrained AE to the adapted AE', Line 18 in Algorithm 1. To do so, the layers in the encoder are set as not trainable and the layers from the latent space and the decoder are trained, see Figure 1. The number of epochs is set to 10, much smaller than those used in the training stage, to avoid the overfitting of the encoder into the new shift, as the main goal of the adaptive strategy is to keep the essence of the previous AE with smooth and fast adaptation. The remaining hyper-parameters of the network are kept constant.

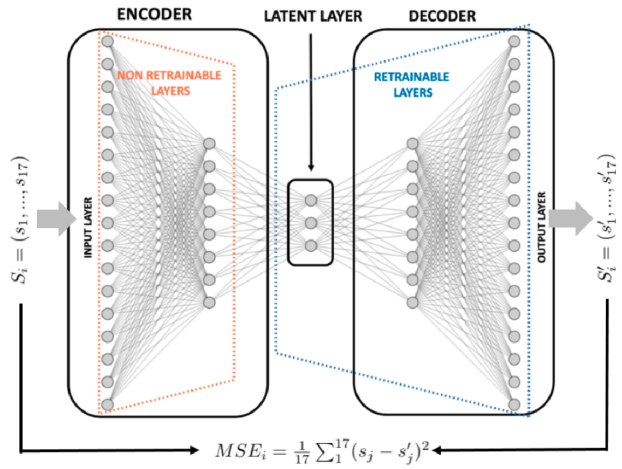


Fig. 1. Architecture of the used autoencoder

C. Anomaly detection methodology

This is a simple and efficient unsupervised anomaly detection method which is based on statistical quantiles that are considered to identify anomalous conditions. The method assumes that the training data belongs to normal working conditions and establishes thresholds over the data distribution to detect abnormal behaviors. Hence, the thresholds are set based on the reconstructions errors obtained by the AE on the training data, X. The 0.95, 0.99 and 0.99 quantiles of the mean squared error distributions, MSE_x, Line 5 in Algorithm 1, are computed. They are referred as minor, major and critical anomalies, respectively. Once the algorithm is deployed, the streaming data reconstruction is computed as shown in Figure 1 and in Line 21 in Algorithm 1. This data is compared against the threshold, Line 22 in Algorithm 1, and the anomalies found are returned, if any, as $\{A_{0.95}, A_{0.99}, A_{0.999}\}$ (see Line 23 in Algorithm 1).

IV. Experiment design

To test the proposed strategy we carried out an experiment based on data from a real industrial process. This data belongs to a manufacturing process of water jet cutting for plastic items in the automotive industry. The experimental dataset contains 17 variables which are the main parameters of the process, e.g.: acceleration, speed, pressure, etc; and it is acquired in time-scale aggregated batches of data streams. They are first aggregated on edge devices next to the machine to later integrate the data from different IIoT sensors within the same timestamps. For training the AE, Line 4 in Algorithm 1, four different references have been used, that altogether form the training dataset X which has a total of 10334 samples. For the testing data a total of 14558 samples are used, belonging to 16 unique references, different from the 4 references used for training. These 16 references are split into 55 manufacturing shifts, as the same references are manufactured in different dates, resulting in more shifts than available references. Each shift has a duration of 4.5 hours on average, ranging from 1 hour to 22.5 hours.

The importance of the W window size, Input 4 in Algorithm 1, has been tested in the experiment by considering three different configurations: 10, 15, and 20. The window size means the number of instances of the new shift used as prototypes and that are buffered to generate the synthetical data that is applied to refit the AE'. The data buffered within the window size must correspond to optimal working conditions. Indeed, when a new reference is manufactured a setup and parameter tuning of the machine is performed, thus resulting in an optimal operational starting point. The KDE bandwidth configuration has been established on the basis of a grid search. It is set to 0.03 since that value leads to best predictive performance. Finally, the MSE values and anomalies are computed, Lines 21-22 Algorithm in 1.

To evaluate the proposed methodology, the performance of the AE model, Line 4 in Algorithm 1, has also been tested in isolation, i.e. without the application of the adaptive strategy. These results establish the baseline that the adaptive strategy must outperform.

V. Experiment results and discussion

The results of the experiments are shown for different window sizes, as specified in section IV. We used the distribution of the reconstruction MSE and the number of anomalies detected as performance metrics to compare our adaptive strategy against the considered baseline. Whenever the adapted AE' is used we refer to it as AL (Adaptive Learning), and when the baseline AE is used we refer to it as FIX. The adaptive strategy and the baseline strategy tested on the test data are compared to the results obtained in the training phase with the offline trained AE.

In Figure 2, the reconstruction error distributions are shown for the different window sizes considered. From the error distribution the impact of the window size can be clearly seen on the Test AL boxplot, which refers to the application of the adaptive learning strategy. The Training FIX and Test FIX boxplots remain constant for different window sizes, as they do not apply the adaptive strategy. The goal of the proposed approach is to detect anomalies within the new shift, which has never

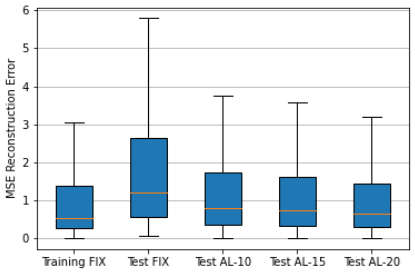


Fig. 2. Mean Squared Error distributions for AE & AE' with training and testing data

been seen before. In a normal operational working conditions, which is the case for our test data, MSE_i distribution from the new shift, should follow as close as possible the distribution MSE_X from the Training FIX. From the results in Figure 2 it can be seen that how the Test AL boxplot is getting closer to the Train FIX boxplot while increasing the window size, reaching the optimal point for values 15 & 20. Additionally, the fixed autoencoder not considering an adaptive strategy leads to great divergence between the Test FIX boxplot and the Train FIX boxplot, as the same pre-trained model is used for every new shift, thus resulting in greater reconstruction errors that trigger a huge amount of anomalies.

Another interesting illustration of the benefits of the proposed adaptive learning strategy is the Q-Q plot, in Figure 3, which compares the ventile values of the Test AL and Test FIX MSE_i distributions against the ventile values of the Train FIX MSE_X distribution. For the sake of interpretation, only the first 18 bins of the ventiles are shown, i.e.: up to the quantile 0.95; as the 19th ventile contains the extreme anomalies that are shown in Table II. It can be considered as a high-resolution representation of the boxplots shown in Figure 2. As the window size increases the AL Q-Q plot gets closer to the optimal curve, which is represented in a straight line, while the FIX Q-Q plot remains static and diverges notoriously from the optimal one. The optimal curve represents the ideal scenario where the MSE_i distributions follow exactly the MSE_X distribution. Again, the optimal window sizes are 15 & 20.

Finally, the anomaly detection task is carried out, as presented in Lines 22-23 in Algorithm 1. As stated earlier, thresholds have been fixed by obtaining the quantiles values from the distribution of the reconstruction error of the training set, MSE_X, resulting from fitting the training set X on the AE model in an offline fashion, Line 5 in Algorithm 1. They are set to 0.95 for a minor anomaly, 0.99 for a major anomaly and 0.999 for a critical anomaly. The anomaly detection results are gathered in Table II, where the expected anomaly rates 1 - Q are shown together with the true percentages of the detected anomalies for the solutions based on the AL strategy implementation and on the FIX AE in isolation.

The differences between the adaptive (AL) and non adaptive (FIX) strategies can be easily compared quantitatively in Table II. We should note that the variation ratios in the FIX algorithm are due to the change in the number of samples, as it is

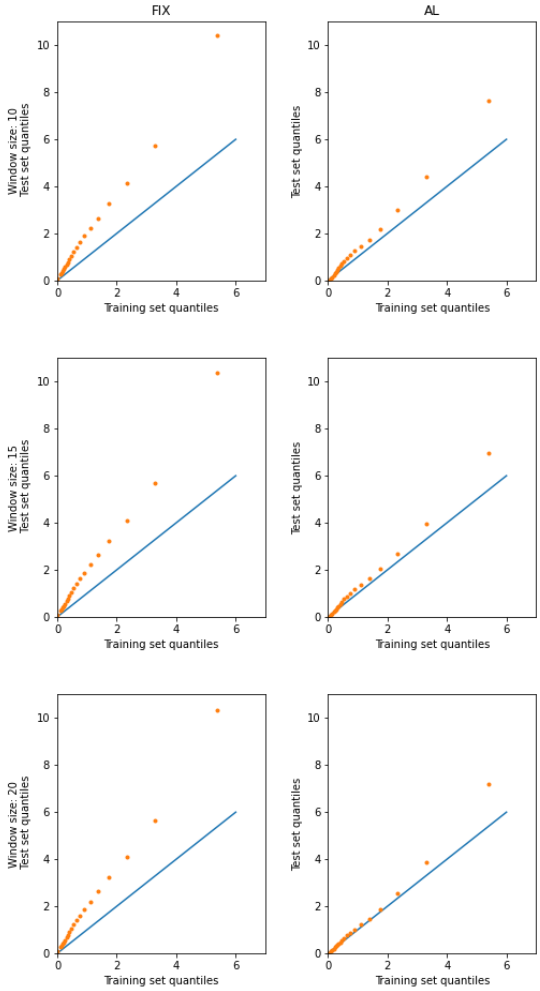


Fig. 3. Q-Q plot - training error distribution vs test error distribution for Test AL and Test FIX. Bins are setup by ventiles (only first 18 bins are shown).

TABLE II
ANOMALY RATIOS FOR DIFFERENT WINDOW SIZES AND QUANTILES

Window Size	Quantile	**Expected**	**FIX**	**AL**
10	0.95	0.050	0.1082	0.0784
10	0.99	0.010	0.0130	0.0130
10	0.999	0.001	**0.0010**	**0.0010**
15	0.95	0.050	0.1068	**0.0697**
15	0.99	0.010	**0.0111**	**0.0111**
15	0.999	0.001	**0.0010**	**0.0010**
20	0.95	0.050	0.1061	0.0731
20	0.99	0.010	0.0113	0.0113
20	0.999	0.001	**0.0010**	**0.0010**

compared with the same data as the AL strategy, which skips the instances buffered in the $refitSample$. Regarding the anomaly detection rates, both the AL strategy and the isolated autoencoder (FIX) are able to detect major and critical anomalies, quantiles 0.99 and 0.999 respectively, with a remarkably good approximation. These results could be expected, as extreme anomalies do not require a high sensitivity, neither

in the training nor in the test datasets. Nevertheless, if we focus on the minor anomaly detection ratios, the quantile 0.95, the difference is substantial, and even though both models commit an error, the best results are achieved by the AL. This is especially significant regarding the window of size 15, with an error rate of 0.0697, meaning that the model will detect 19.7% more anomalies than expected. Instead, if the fixed AE was to be applied, this number would increase to more than 50%, thus making a huge difference between applying or not applying the proposed strategy.

VI. Conclusion and Future Research

An adaptive learning strategy for unsupervised anomaly detection under data shift scenarios in industrial processes has been presented. The implementation of this strategy is expected to bring major benefits in the industrial mass production lines, where the increasing demand for the manufacturing of more personalized products requires more flexible and adaptive solutions. The ultimate goal of the presented strategy is to reduce the number of quality issues by detecting and anticipating the anomalies to efficiently implement preventive action plans beforehand. The capability of the strategy in detecting minor anomalies within the industrial processes is very important, as it is a key factor to avoid the evolution from minor defects to critical issues and production nonconformities. Our adaptive learning strategy has shown to be effective in detecting minor anomalies under data shift scenarios, and has outperformed the baseline scenario where no strategy is applied by a factor of 2.5. The fixed AE, not considering an adaptive strategy, has resulted in a much higher anomaly rate than expected. This leads to a bad outcome, firstly because of the lack of confidence within the detection algorithm, and hence to inaction when the system detects a minor anomaly. Therefore, as seen in Section V, the importance of the similarity of the reconstruction error distribution from the training and the test sets is crucial for the anomaly detection, as a high divergence will lead to undesirable results.

In conclusion, the adaptive learning strategy introduced in this paper and tested into a real industrial case study is a valid solution when unsupervised data and data shift occur simultaneously, and it gives better results than the static methodology. The discussed real case represents one of many industrial processes that can be found in industrial factories and due to the autoencoders' self-learning capabilities, it could be easily implemented in other domains, either industrial or of any kind, making the strategy flexible and easy to deploy.

As a future work, an interesting research line would be to incorporate explainable artificial intelligence (XAI) methodologies to the proposed strategy. This would remarkably enrich the current research by linking the detected anomalies with a root-cause analysis. Moreover, more research has to be done in the adaptive learning process, involving the machine operators in the process by means of human-in-the-loop strategies.

References

[1] T. Hu, T. Tang, R. Lin, M. Chen, S. Han, and J. Wu, "A simple data augmentation algorithm and a self-adaptive convolutional architecture for few-shot fault diagnosis under different working conditions," *Measurement*, vol. 156, p. 107539, 2020.

[2] Z. Ge, Z. Song, S. X. Ding, and B. Huang, "Data mining and analytics in the process industry: The role of machine learning," *IEEE Access*, vol. 5, pp. 20 590–20 616, 2017.

[3] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep learning*. MIT press Cambridge, 2016, vol. 1, no. 2.

[4] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, "A survey on deep transfer learning," in *International conference on artificial neural networks*. Springer, 2018, pp. 270–279.

[5] A. Diez-Olivan, P. Ortego, J. Del Ser, I. Landa-Torres, D. Galar, D. Camacho, and B. Sierra, "Adaptive dendritic cell-deep learning approach for industrial prognosis under changing conditions," *IEEE Transactions on Industrial Informatics*, 2021, in press.

[6] A. Diez-Olivan, J. Del Ser, D. Galar, and B. Sierra, "Data fusion and machine learning for industrial prognosis: Trends and perspectives towards industry 4.0," *Information Fusion*, vol. 50, pp. 92–111, 2019.

[7] D. Sik and J. Levendovszky, "Detecting outliers and anomalies to prevent failures and accidents in industry 4.0," in *2020 11th IEEE International Conference on Cognitive Infocommunications (CogInfoCom)*. IEEE, 2020, pp. 000 355–000 358.

[8] F. Psarommatis, G. May, P.-A. Dreyfus, and D. Kiritsis, "Zero defect manufacturing: state-of-the-art review, shortcomings and future directions in research," *International Journal of Production Research*, vol. 58, no. 1, pp. 1–17, 2020.

[9] M. Ghobakhloo, "Industry 4.0, digitization, and opportunities for sustainability," *Journal of Cleaner Production*, vol. 252, p. 119869, 2020.

[10] V. C. Müller and N. Bostrom, "Future progress in artificial intelligence: A survey of expert opinion," in *Fundamental issues of artificial intelligence*. Springer, 2016, pp. 555–572.

[11] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," California Univ San Diego La Jolla Inst for Cognitive Science, Tech. Rep., 1985.

[12] Y. Wang, H. Yao, and S. Zhao, "Auto-encoder based dimensionality reduction," *Neurocomputing*, vol. 184, pp. 232–242, 2016.

[13] A. Krizhevsky and G. E. Hinton, "Using very deep autoencoders for content-based image retrieval." in *ESANN*, vol. 1, 2011, p. 2.

[14] M. Sakurada and T. Yairi, "Anomaly detection using autoencoders with nonlinear dimensionality reduction," in *Proceedings of the MLSDA 2014 2nd Workshop on Machine Learning for Sensory Data Analysis*, 2014, pp. 4–11.

[15] L. Theis, W. Shi, A. Cunningham, and F. Huszár, "Lossy image compression with compressive autoencoders," *arXiv preprint arXiv:1703.00395*, 2017.

[16] J. Su, D. Xiong, B. Zhang, Y. Liu, J. Yao, and M. Zhang, "Bilingual correspondence recursive autoencoder for statistical machine translation," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 2015, pp. 1248–1258.

[17] Y. Kawachi, Y. Koizumi, and N. Harada, "Complementary set variational autoencoder for supervised anomaly detection," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 2366–2370.

[18] T. Chen, X. Liu, B. Xia, W. Wang, and Y. Lai, "Unsupervised anomaly detection of industrial robots using sliding-window convolutional variational autoencoder," *IEEE Access*, vol. 8, pp. 47 072–47 081, 2020.

[19] J. Quiñonero-Candela, M. Sugiyama, N. D. Lawrence, and A. Schwaighofer, *Dataset shift in machine learning*. Mit Press, 2009.

[20] J. Gama, P. Medas, G. Castillo, and P. Rodrigues, "Learning with drift detection," in *Brazilian symposium on artificial intelligence*. Springer, 2004, pp. 286–295.

[21] I. Žliobaitė, "Learning under concept drift: an overview," *arXiv preprint arXiv:1010.4784*, 2010.

[22] D. P. Kingma and M. Welling, "An introduction to variational autoencoders," *arXiv preprint arXiv:1906.02691*, 2019.

[23] M. Baccouche, F. Mamalet, C. Wolf, C. Garcia, and A. Baskurt, "Spatio-temporal convolutional sparse auto-encoder for sequence classification." in *BMVC*, vol. 1, 2012, p. 12.

[24] X. Hou, L. Shen, K. Sun, and G. Qiu, "Deep feature consistent variational autoencoder," in *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2017, pp. 1133–1141.

Train Route Planning as a Multi-Agent Path Finding Problem*

*Note: The full contents of this paper have been published in the volume Lecture Notes in Artificial Intelligence 12882 (LNAI 12882)

Mauricio Salerno, Yolanda E-Martín, Raquel Fuentetaja, Alba Gragera, Alberto Pozanco, Daniel Borrajo
Universidad Carlos III de Madrid
Madrid, Spain
{msalerno,agragera,apozanco}@pa.uc3m.es, {yescuder,rfuentet}@inf.uc3m.es, dborrajo@ia.uc3m.es

***Abstract*—The train routing and timetabling problem consists of setting routes and schedules of a set of vehicles given their initial timetables and a railway network. The number of vehicles, the complexity and limited capacity of the railway network, and the time constraints make this problem difficult to solve. In this paper, we model this problem as a Multi-Agent Pathfinding problem, and propose a Conflict-Based Search approach to solve it. In our approach, we consider the complex properties found in this scenario such as continuous time, agents that function as convoys of arbitrary length, arbitrary action duration, and railway networks to find a solution. We analyze and discuss our approach explaining the main difficulties and evaluate it on several scenarios.**

***Index Terms*—Train routing and timetabling, Multi-agent path finding, Heuristic search.**

DOI: https://doi.org/10.1007/978-3-030-85713-4_23

Asynchronous Vector Iteration in Multi-Objective Markov Decision Processes*

*Note: The full contents of this paper have been published in the volume Lecture Notes in Artificial Intelligence 12882 (LNAI 12882)

Ekaterina Sedova, Lawrence Mandow, José-Luis Pérez-de-la-Cruz
Dpt. Lenguajes y Ciencias Comp.
Univ. de Málaga, Andalucía Tech
Malaga, Spain
ekaterina.sedova@tum.de, {lawrence,perez}@lcc.uma.es

***Abstract*—This paper presents new algorithms to solve Multi-Objective Markov Decision Processes (MOMDPs). Namely, we present Multi-objective Dynamic Programming variants of Value Iteration such that the values for every state are updated in some heuristic order. The performance of these algorithms is evaluated applying them to benchmark problems with two and three objectives.**

DOI: https://doi.org/10.1007/978-3-030-85713-4_13

A Similarity Measure of Gaussian Process Predictive Distributions*

*Note: The full contents of this paper have been published in the volume Lecture Notes in Artificial Intelligence 12882 (LNAI 12882)

Lucía Asencio Martín, Eduardo C. Garrido-Merchán
Universidad Autónoma de Madrid
Madrid, Spain
lucia.asencio@estudiante.uam.es, eduardo.garrido@uam.es

***Abstract*—Some scenarios require the computation of a predictive distribution of a new value evaluated on an objective function conditioned on previous observations. We are interested on using a model that makes valid assumptions on the objective function whose values we are trying to predict. Some of these assumptions may be smoothness or stationarity. Gaussian process (GPs) are probabilistic models that can be interpreted as flexible distributions over functions. They encode the assumptions through covariance functions, making hypotheses about new data through a predictive distribution by being fitted to old observations. We can face the case where several GPs are used to model different objective functions. GPs are non-parametric models whose complexity is cubic on the number of observations. A measure that represents how similar is one GP predictive distribution with respect to another would be useful to stop using one GP when they are modelling functions of the same input space. We are really inferring that two objective functions are correlated, so one GP is enough to model both of them by performing a transformation of the prediction of the other function in case of inverse correlation. We show empirical evidence in a set of synthetic and benchmark experiments that GPs predictive distributions can be compared and that one of them is enough to predict two correlated functions in the same input space. This similarity metric could be extremely useful used to discard objectives in Bayesian many-objective optimization.**

DOI: https://doi.org/10.1007/978-3-030-85713-4_15

Towards Fairness in Classification: Comparison of Methods to Decrease Bias*

*Note: The full contents of this paper have been published in the volume Lecture Notes in Artificial Intelligence 12882 (LNAI 12882)

Maitane Martínez-Eguiluz, Oier Irazabal-Urrutia, Olatz Arbelaitz-Gallego
Department of Computer Architecture and Technology
University of the Basque Country (UPV/EHU)
Donostia, Spain
{mmartinez149,oirazabal003}@ikasle.ehu.eus, olatz.arbelaitz@ehu.eus

***Abstract*—It is said that with great power comes great responsibility. Nowadays, we rely on machine learning systems to make decisions. Unfortunately these systems suffer from algorithmic biases; they often produce results that are systemically prejudiced due to erroneous assumptions in the machine learning process. Consequently these systems can contribute to increase biases in society and this is something we should avoid undoubtedly. The importance of the topic and the effect it has in the society has made it become an important research topic during the last years giving rise to different solutions. In this work, we selected three state-of-the-art techniques, decoupled classifiers, fairness constraints and adversarial learning, that claim to reduce bias in machine learning algorithms and compared their performance over different databases and fairness evaluation metrics. The obtained results show that there is no system performing the best in all aspects and databases but gives some hints to select the best option according to the objective.**

***Index Terms*—Machine learning, Algorithmic bias, Fairness, Evaluation.**

DOI: https://doi.org/10.1007/978-3-030-85713-4_9

Modeling Administrative Discretion Using Goal-Directed Answer Set Programming*

*Note: The full contents of this paper have been published in the volume Lecture Notes in Artificial Intelligence 12882 (LNAI 12882)

Joaquín Arias, Mar Moreno-Rebato, José A. Rodríguez-García, Sascha Ossowski
CETINIA
Universidad Rey Juan Carlos
Madrid, Spain
{joaquin.arias,mar.rebato,joseantonio.rodriguez,sascha.ossowski}@urjc.es

***Abstract*—Automated legal reasoning and its application in smart contract is getting interest. In this context, ethical and legal concerns make it necessary for automated reasoners to *justify* in human-understandable terms the advice given. Logic Programming, specially Answer Set Programming, has a rich semantics and has been used to very concisely express complex knowledge. However, modelling vague concepts such as *ambiguity* and *discretion* cannot be expressed in top-down execution models based on Prolog, and in bottom-up execution models based on ASP the justifications are incomplete and/or not scalable. We propose to use s(CASP), a top-down execution model for predicate ASP, to model ambiguity and discretion following a set of patterns. We have implemented a framework, called s(LAW), to model, reason, and justify the applicable legislation and validate it by translating (and benchmarking) the criteria for the admission of students in public centers established by the "Comunidad de Madrid".**

***Index Terms*—Answer Set Programming, Goal-Directed, Ambiguity, Administrative Discretion.**

DOI: https://doi.org/10.1007/978-3-030-85713-4_25

Automatic Generation of Interrelated Organisms on Virtual Environments*

*Note: The full contents of this paper have been published in the volume Lecture Notes in Artificial Intelligence 12882 (LNAI 12882)

Santiago Pacheco, Nicolás Ottonello, Sergio Nesmachnow
Universidad de la República
Uruguay
{santiago.pacheco,nicolas.ottonello,sergion}@fing.edu.uy

***Abstract*—This article presents a novel method for generating multiple species of organisms on virtual environments, with interrelated behaviors. The proposed system (AGIO) strives to keep a balance between the exploration of diverse species, the evolution of interesting and connected behaviors, and the control of the user over the generated organisms. Experimental results of a C++ implementation proves that AGIO is capable of generating different species and basic animal conducts, while keeping behaviors of the organisms interrelated. Multiple relations between different species were observed on the generated organisms. Furthermore, the performance of the organisms was comparable to that of a human player.**

***Index Terms*—Neuroevolution, A-Life, Video games**

DOI: https://doi.org/10.1007/978-3-030-85713-4_12

An analysis of the indexes measuring the agreement of a profile of rankings

Noelia Rico*, Camino R. Vela*, Irene Díaz*
*Department of Computer Science, University of Oviedo
Emails: [noeliarico,crvela,sirene]@uniovi.es

***Abstract*—The Kemeny method is a well known procedure to solve ranking aggregation problems based on selecting as consensus ranking the one that minimises the distance to the rankings given by the voters. In this work we review and propose some indexes to characterise the difficulty of providing the kemeny ranking in terms of the execution time taken by an exact algorithm to solve the problem. The behaviour of the different indexes is studied, highlighting that each one partially explains the difficulty of finding the Kemeny ranking but none of them allows us to completely characterise this difficulty.**

***Index Terms*—ranking aggregation, index of difficulty, agreement, computation time**

I. Introduction

The need of making a decision based on the preferences of several voters expressed in the form of rankings emerges in a variety of contexts. In particular, the problem of obtaining the *consensus ranking* that best summarises the rankings provided by several voters has been deeply studied in the field of social choice theory. One of the first contributors to this field was Condorcet, who stated that if the preferences given by the voters are expressed in the form of rankings, the only information that should be used in order to achieve a consensus ranking is that given by the pairwise comparisons between the alternatives. Using this fact, he considered the best alternative to be the one beating all other alternatives in a pairwise comparison by a majority of the votes.

In case the goal is to establish a winning ranking, the majority criterion is often applied, that is, an alternative should be ranked at a better position than another alternative in the final ranking whenever the former defeats the latter by more than half of the votes. Unfortunately, the majority relation obtained from the pairwise comparisons is not necessarily transitive (even if all the rankings in the profile are transitive) and consequently cycles of preferences might occur among three or more alternatives, which is known in the literature as the *voting paradox*.

In the last century, several attempts to fully axiomatize Condorcet's approach for solving the problem in the presence of cycles have been made by different authors [1], [2]. All these proposed methods for the aggregation of rankings are usually referred to as *Condorcet methods*, as the ranking they provide coincides with the Condorcet ranking if it exists [3].

As one of the most prominent such examples, the method proposed by Kemeny selects as the consensus ranking the one that minimizes the distance to the rankings given by the voters [4]. Therefore, after computing the distance of each possible ranking to the profile of rankings, the Kemeny method selects the ranking that is *the closest* to all rankings given by the voters. This method is known to be the only one that is neutral, consistent and Condorcet [5]. Despite its good properties, the problem of finding the Kemeny ranking is known to be NP-hard [6]. Thus, it is not suitable most real-life problems, especially in presence of time constraints.

The computation of ranking aggregation methods [7] has gained attention in recent years, as they arise quite commonly in real-life problems. Recently, Azzini and Munda [8] introduced an exact algorithm that greatly improved the execution time in comparison to the previously available methods. The time required to solve the problem obviously depends on the algorithm used, but even for the same algorithm this time widely varies for profiles of rankings with the same number of alternatives and voters [9], [10]. This difference in execution time could be attributed to the disagreement between the voters. However, how to measure this disagreement between the voters is not trivial and it does become a challenging task to measure the difficulty of finding a solution with respect to the disagreement for the given profile of rankings and to analyse the impact that this disagreement has on the execution time. In fact, the proper definition of *difficulty* in this context is not standard. In this work it is defined as the amount of time required by the implementation of the algorithm proposed in [8] to find the solution of the Kemeny problem.

The aim of this work is to characterize a profile of rankings in order to answer the question *how difficult is to find the solution in terms of execution time* by reviewing and proposing indexes that allow to characterize the profile of rankings based on its properties. The remainder of this paper is organized as follows. Section II presents the Kemeny method. Section III introduces the latest exact algorithm in the literature and discusses why some profiles may be more difficult to solve than others in terms of computational time. The indexes already existing in the literature are presented in Section IV and, additionally, new indexes are proposed in Section V. Some conclusions and future lines of research are outlined in Section VII.

II. The Kemeny method

Consider a set of n alternatives $\mathcal{A} = \{a_1, \ldots, a_n\}$. Preferences over $\mathcal{A}$ are expressed in the form of a ranking, which is a complete, reflexive and transitive (but not necessarily antisymmetric) relation such that, for every pair of alternatives

$a_i, a_j \in \mathcal{A}$, a strict ($a_i \succ a_j$ or $a_j \succ a_i$) or equivalence ($a_i \sim a_j$) relation is defined. The list of rankings given by m different voters over the set of n alternatives is called the profile of rankings, denoted by π_m^n. The alternatives in $\mathcal{A}$ may be compared in a pairwise fashion by using a matrix $\mathcal{O}$ of dimension $n \times n$, known as the outranking matrix [11]. Each element $o_{ij} \in \mathbf{O}$ represents the number of times that the alternative a_i is preferred over the alternative a_j. The value of the element o_{ij} is obtained from π_m^n by adding 1 point every time that $a_i \succ a_j$ in a ranking of the profile and 0.5 points every time that $a_i \sim a_j$. Therefore, for each pair of alternatives a_i and a_j with $i \neq j$, it holds that $o_{ij} + o_{ji} = m$. By definition, all the elements of the diagonal are set to 0.

The Kemeny method looks for the ranking(s) r that minimises a distance $\delta(r, \pi_m^n)$ to the profile. This is often understood under the equivalent definition of solving the problem of maximising the agreement between voters. Under this definition, these rankings can be computed using the outranking matrix by seeking the rankings with maximum agreement score σ such that

$$\sigma(s, \pi_m^n) = \sum_{i=1}^{n} \sum_{j=1}^{n} o_{ij} \cdot x_{ij}, \tag{1}$$

where $x_{ij} = 1$ if $a_i \succ^s a_j$ and 0 otherwise.

In order to determine the Kemeny ranking s^*, it is necessary to explore all the possible $n!$ (antisymmetric) rankings that can be obtained from permuting the set of n alternatives and subsequently to find the ranking that maximises the value of sigma σ. Thus, the Kemeny problem can be formalised as

$$s^* = \underset{s}{\operatorname{argmax}}\, \sigma(s, \pi_m^n)\,. \tag{2}$$

III. The difficulty of the profile of rankings

Despite having many desirable properties, the fact that the Kemeny method requires the exploration of the $n!$ possible winning rankings makes it intractable and therefore impossible to use in many real problems. Additionally, it is known that the time to obtain the Kemeny ranking not only depends on the number of alternatives but also on some issues related to the profile structure such as the distribution of the votes over the alternatives, which is related to the disagreement between voters. However, how to measure this disagreement and its impact on the difficulty of solving the problem is not a trivial problem. Although some authors have proposed indexes [9], there is not a unique measure that allows to know how difficult the Kemeny problem is given a profile of rankings.

Furthermore, the execution time is conditioned by the algorithm chosen to solve the problem. For example, algorithms that find an optimal solution based on search trees that prune the tree according to a lower bound depend on where the optimal solution is placed on the tree.

In [8] an exact algorithm is proposed that greatly reduces the execution time of the previously existing algorithms. This time reduction is achieved by taking into account that the alternative at the top position $a_i \in \mathcal{A}$ satisfies $\sum_{j=1}^{n} o_{ij} \geq \sum_{j=1}^{n} o_{ji}$. This necessary condition avoids the exploration of many of the rankings as potential solutions, thus considerably reducing the execution time in comparison to other algorithms. For the sake of simplicity, in the remaining of this paper α_i represents the following truth value

$$\alpha_i = \left(\sum_{j=1}^{n} o_{ij} \geq \sum_{j=1}^{n} o_{ji} \right)\,. \tag{3}$$

When the alternative a_i is preferred over the remaining alternatives in $\mathcal{A}$ at least as many times as the other alternatives are preferred over a_i, then $\alpha_i = True$. As a consequence, the alternative a_i at the first position of the ranking always satisfies $\alpha_i = True$. Using the previous condition, the authors define the `Mork-Exact` algorithm to solve the Kemeny problem, which is outlined as follows:

Step 1. In an outranking matrix with ($n \geq 3$) choose all alternatives with $\alpha_i = True$.

Step 2. Pick up the first of those alternatives. Add the alternative to the ranking and delete its corresponding row and column.

Step 3. Repeat *Step 1* and *Step 2* recursively. The stop condition of the recursive process is to find a matrix of dimension 2×2, in this case add to the ranking the alternative with largest non-diagonal value first and then the other.

Step 4. Compute the total score for all the rankings obtained. Delete those with total score lower than the maximum one.

IV. Existing indexes to measure profile disagreement

The identification of the characteristics of the profile of rankings that make it difficult to find the solution of the Kemeny method is a challenging task. In the literature some authors have pointed out the different execution times obtained [9], [10] and the importance of defining indexes that allow to parametrize algorithms [12]. In this section, the previously defined *indexes* that can help to characterise some of the properties of a profile of rankings are presented. In the remaining of this paper τ_n (resp. τ_m) represents the number of pairwise comparisons of the n alternatives (resp. m voters), that is $\tau_i = \frac{i(i-1)}{2}$, where $i = \{n, m\}$.

A. Average Kendall distance

The average Kendall distance [13] between the rankings in the profile is proposed by [10] as an index for measuring the difficulty to determine a consensus ranking. It is defined in terms of $\mathcal{O}$ as

$$\mu_1 = \frac{1}{\tau_m} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} o_{ij} \cdot o_{ji}\,.$$

If the m voters of a profile of rankings express the same preference, the distance between each pair of rankings is 0 and that would be the value of μ_1. On the other hand, the worst scenario, i.e, the maximum disagreement, is produced with

two voters, where one of the voters expresses one ranking and the other one expresses the opposite ranking. This case would lead to a distance τ_n. However notice that, when $m > 2$, the value of the index is given by $\frac{m^2}{2*(m^2-m)} < 1$ and therefore $\mu_1 < \tau_n$. If m is odd, it can be deduced with an analogous reasoning that $\mu_1 < \tau_n$.

B. Range of positions taken by an alternative

As each alternative $a_j \in \mathcal{A}$ can take different positions in a ranking, it makes sense to consider that if for an alternative the range of its positions is large, then there is more disagreement between the voters' opinion in relation to that alternative.

This range can take a value in the interval of natural numbers $[0, n-1]$, where the value 0 is obtained when an alternative is always ranked at the same position in all the rankings and the value $n-1$ when an alternative is such that $r_i(a_j) = 1$ in at least one ranking and $r_k(a_j) = n$ in at least another ranking, where $r_i(a_j) = l$ denotes the fact that the position of $a_j \in \mathcal{A}$ in the ranking $r_i \in \pi_m^n$ is l. If all the voters provide the same ranking, the maximum range for all alternatives is 0. Based on the position range, three different indexes have been considered [10].

1) Maximum range: Intuitively, a high value of this index means that there is at least one alternative for which the voters do not have a clear opinion. It is computed as follows:

$$\mu_2 = \max(|r_i(a_k) - r_j(a_k)|), \ \forall r_i, r_j \in \pi_m^n, \ \forall a_k \in \mathcal{A}.$$

Note that although there is not a direct influence of m, it is natural to think that when the number of voters increases, it becomes more likely to have a greater maximum range.

2) Minimum range: In the same line, we consider also the minimum range, which can be defined as follows:

$$\mu_3 = \min(|r_i(a_k) - r_j(a_k)|), \ \forall r_i, r_j \in \pi_m^n, \ \forall a_k \in \mathcal{A}.$$

When μ_3 is low, then there is at least one alternative in the profile for which the voters have a clear opinion about.

3) Average range: It measures the average range of the alternatives in the rankings of the profile:

$$\mu_4 = \frac{1}{n} \sum_{a_k \in \mathcal{A}} \max_{r_i \in \pi_m^n} (r_i(a_k)) - \min_{r_j \in \pi_m^n} (r_j(a_k)) \, .$$

This average also belongs to the interval $[0, n-1]$.

C. Bounds to the optimal cost

To compute the score from a ranking of n alternatives to a profile of rankings, it is necessary to sum a total of τ_n elements of the outranking matrix $\mathcal{O}$, as defined in Eq. (1). The bounds that this score can take vary depending on how the votes are distributed over the alternatives [9].

$$\mu_5 = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \max(o_{ij}, o_{ji}) - \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \min(o_{ij}, o_{ji}) \, .$$

D. Intransitivity

An unordered triplet of alternatives (a_i, a_j, a_k) is called intransitive if all three of o_{ij}, o_{jk}, o_{ki} are simultaneously greater than half of the voters.If the profile of rankings presents a Condorcet ranking, then there are no cycles between the alternatives and therefore it is straightforward to obtain the consensus ranking. As the number of cycles increases, the solution is further from being the Condorcet ranking. The number of cycles can be formally defined as

$$\mu_6 = \frac{1}{3} \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{n} x_{ijk} \cdot y_{ijk} \ ,$$

with

$$x_{ijk} = \begin{cases} 1 & \text{, if } i \neq j \wedge i \neq k \wedge j \neq k, \\ 0 & \text{, otherwise} \end{cases}$$

and y_{ijk} is true if $o_{ij} > h \wedge o_{jk} > h \wedge o_{ki} > h$, with h representing half of the voters. More precisely, $h = \frac{m}{2}$ if m is even and $h = \frac{m-1}{2}$ if it is odd.

V. New indexes for measuring the profile disagreement

A. Indexes based on pairwise voter opinions

The rankings in the profile could be simplified into a pairwise comparison of the alternatives such that, for each pair $a_i, a_j \in \mathcal{A}$, only the total number of voters preferring one over the other one is taken into account, not considering their orders in relation to the remaining alternatives in the profile. Following this approach, the *margin* of the voters preferring one alternative a_i over another one a_j is $|o_{ij} - o_{ji}|$, and it indicates the global preference expressed in the profile for one of the two alternatives over the other one. Different issues related to these margins may influence the difficulty of determining a solution. For example, the average value of the margin is equivalent to measure the bounds to the optimal cost due to the constant sum property of $o_{ij} + o_{ji} = m$. The indexes proposed below try to capture more aspects related to the pairwise comparison of the alternatives in the profile.

1) Mode margin: If the most common margin has a large value then the agreement should be greater at least for some pairs, although they may still disagree in others. Consider the margin set $\mathbb{M} = \{M_{ij} = |o_{ij} - o_{ji}| \mid 1 \leq i < j \leq n\}$, then

$$\mu_7 = \underset{M \in \mathbb{M}}{\operatorname{argmax}} \, \#\{(i,j) \mid 1 \leq i < j \leq n \, , \, M = M_{ij}\} \, .$$

Therefore, this index takes one of the possible values of the indexes such that $\mu_7 \in [x, m]$, with $x = m \bmod 2$. Notice that for an even number of voters the possible values of the margin are even, whereas for an odd number of voters the values of the margin are odd.

2) Number of different margins: Margin diversity may also influence the difficulty of finding the solution. However, the same margin diversity can represent different situations. For example, the number of different margins equals 1 if there is a unique ranking in the profile but also when all the alternatives are tied in the pairwise comparison. Although this is not an intuitive measure, it is worth to further explore it, as in relation with other indexes this might capture profiles that require more time to solve. For example, situations when there is a low number of different margins and the average margin is close to the mean could imply that more combinations should be explored before reaching a solution. The index can be defined as the cardinality of $\mathbb{M}$. Notice that the number of different possible margins depends on the value of m. However, since there are only τ_n pairwise comparisons of alternatives, this index cannot exceed that limit, therefore taking a value in the interval $[1, \tau_n]$:

$$\mu_8 = \#(\mathbb{M}) \,,$$

3) Frequency of the minimum margin: This index measures the number of times that the voters have the maximum possible disagreement for a pair of alternatives, meaning that half (or almost half) of them prefer one over the other one, whereas the other half has the opposite preference. This pairwise maximum disagreement can be represented by the lowest margin ($margin_{min}$), which equals 0 when m is even and equals 1 when m is odd. The proposed index counts the number of times that $margin_{min}$ appears in $\mathcal{O}$, which can be a value in the interval $[0, \tau_n]$, and it is formally defined by

$$\mu_9 = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \gamma_{ij} \,,$$

with $\gamma_{ij} = 1$ if $|o_{ij} - o_{ji}| = margin_{min}$ and 0 otherwise.

B. Indexes based on rowsums

1) Based on the Borda ranking: Borda defined a method [14] to determine a consensus ranking that rewards each alternative with a point every time that other alternative is ranked at a worse position, obtaining a score for each alternative. If an alternative is tied with another alternative, then it assigns half a point. Then, the consensus ranking is obtained by ranking the alternatives in decreasing order according to their obtained score. As the $i-$th row of $\mathcal{O}$ contains the number of times that a_i is preferred over the other alternatives, the Borda score is $s(a_i) = \sum_{j=1}^{n} o_{ij} \; \forall a_i$.

In [9], the authors propose the Kemeny distance (δ) from the Borda ranking to the profile of rankings as a measure of disagreement of the profile. Notice that this output ranking may not be a strict order, as alternatives may have obtained the same score and then be tied at the same position when the output ranking is obtained. Since we are here interested in antisymmetric rankings, a linear extension of the Borda ranking b is considering by just sorting lexicographically the tied elements. Therefore, this index can be computed as

$$\mu_{10} = \delta(b, \pi_m^n) \,.$$

2) Number of overall preferred alternatives: It counts the number of alternatives in the profile that are preferred by the voters more times in comparison to other alternatives than the amount of times that other alternatives are preferred over them. In terms of the outranking matrix, these are the alternatives whose rowsum is greater than its column sum. That means that the number of times they are preferred over other alternatives is larger than the number of times other alternatives are preferred over them. It is defined as

$$\mu_{11} = \sum_{a_i \in \mathcal{A}} \alpha_i \,,$$

with α_i defined as in Eq. (3). Note that $\mu_{11} \in [1, n]$.

3) Standard deviation of rowsum: The standard deviation of the outranking rowsums ($\mu_{12} = sd(\sum_{j=1}^{n} o_{ij})$) can be used as index for characterizing the profile of rankings.

C. Indexes based on the distribution of the Condorcet ranking

The *Condorcet ranking* is a strict order, which ensures that each alternative is preferred to each of the alternatives ranked at a worse position by the majority of the voters. Although this ranking may not exist, if a profile of rankings has a Condorcet ranking, the outranking matrix must be transitive. Therefore, the Condorcet ranking can be determined by counting how many times each alternative $a_i \in \mathcal{A}$ is preferred over other alternatives. The Boolean outranking matrix $\mathbf{B}$ is obtained from the outranking matrix $\mathcal{O}$, such that $b_{ij} = 1$ if $o_{ij} > h$ and $b_{ij} = 0$ otherwise. Consider

$$\beta_i = \sum_{j=1}^{n} b_{ij} \,, \tag{4}$$

which represents the number of times that the alternative a_i is preferred by at least half of the voters over another alternative. Each rowsum of $\mathbf{B}$ takes a different integer in the interval $[0, n-1]$ if there exists a Condorcet ranking. Thus, when the alternatives are sorted in descending order according to their value β_i, the Condorcet ranking is obtained. Let us denote by $\vec{\beta}$ the vector of n elements containing all the values β_i corresponding to each alternative. When there is not a Condorcet ranking, it still holds that $\beta_i \in [0, n-1]$ but not all the values appear in the vector and some of them will be repeated. To study the distribution of $\vec{\beta}$ as a measure of profile disagreement, we consider the standard deviation ($\mu_{13} = sd(\vec{\beta})$) to measure $\vec{\beta}$ spread, the average (μ_{14}), the median (μ_{15}) and the difference between the mean and the median (μ_{16}), as this last value is ensured to be 0 for those profiles that have a Condorcet ranking.

VI. Results

In this section the behaviour of the proposed indexes with regard to the execution time when the Kemeny ranking is computed using the algorithm presented in Section III, is studied. To perform such study, a synthetic dataset of profiles of rankings has been generated.[1]. This dataset contains profiles

[1]This dataset is available at http://github.com/noeliarico/consensus_benchmark

of rankings for $n = 10$ alternatives with different number of voters $m \in \{10, 50, 340, 890\}$ and also the same numbers plus one, in order to check the behaviour of the profiles depending on the nature of the number of voters (even or odd). For each combination of (n, m), 2000 different profiles of rankings are included. Thus, the dataset contains 16000 different profiles of rankings.

All the indexes listed in this paper have been applied to this dataset. Execution time to obtain the Kemeny ranking associated to each profile is also gathered. Fig. 1 exemplifies the behaviour of the indexes already defined in the literature (see Section IV), separating the profiles by its number of voters. For each number of voters the number of voters plus one is also considered (in each graph, triangles represent the original even value of m and circles represent $m+1$) to study the behaviour of ties. In each figure the X-axis represents the value of one of the indexes and the Y-axis represents the execution time in seconds. The X-axis has been liberated for the indexes depending on the number of voters (μ_5 and μ_{10}) in order to show the behaviour of them in relation to the number of voters.

Fig. 2 shows the indexes based on the pairwise comparison of the votes. The frequency of the minimum margin (μ_9) is reduced when the number of voters increases and furthermore, it seems to take its highest peaks within each number of voters for profiles with higher execution time.

In Fig. 3 the behaviour of the indexes based on the rowsum comparison of the votes is shown. On the contrary, the standard deviation of the rowsum (μ_{12}) has generally lower values for profiles that require more execution time. This does not feel intuitive, but in computational terms, the fact that the standard deviation is low may indicate that all the possible solutions have similar costs, therefore making the profiles with higher values of this index quicker to solve as they would allow to discard more tentative solutions faster.

Fig. 4 shows the indexes based on the vector representing the Condorcet distribution. The peak on the right parts of the average (μ_{14}) is created due to the nature of the number of voters. When m is an odd value, for each pair of alternatives there must be one that is preferred over the other one. Therefore, the sum of the elements in $\vec{\beta}$ is constant and consequently also the average. This number varies for an even number of voters, where the sum of $\vec{\beta}$ for two alternatives may be tied in the pairwise comparison, not adding anything to their corresponding β_i.

From Fig. 1 to Fig. 4, it is possible to deduce that all execution time decreases when the number of voters increases. This is a general trend that all the indexes show, independently of their value. It may suggest that indexes having a similar behaviour may be good indicators of the difficulty. On the other hand, note that highest computation time is produced in general when m is low or when m is even (most of peaks are associated to an even number of voters, triangles in the figures). In addition, some indexes tend to a slower range of values when the number of voters increases.

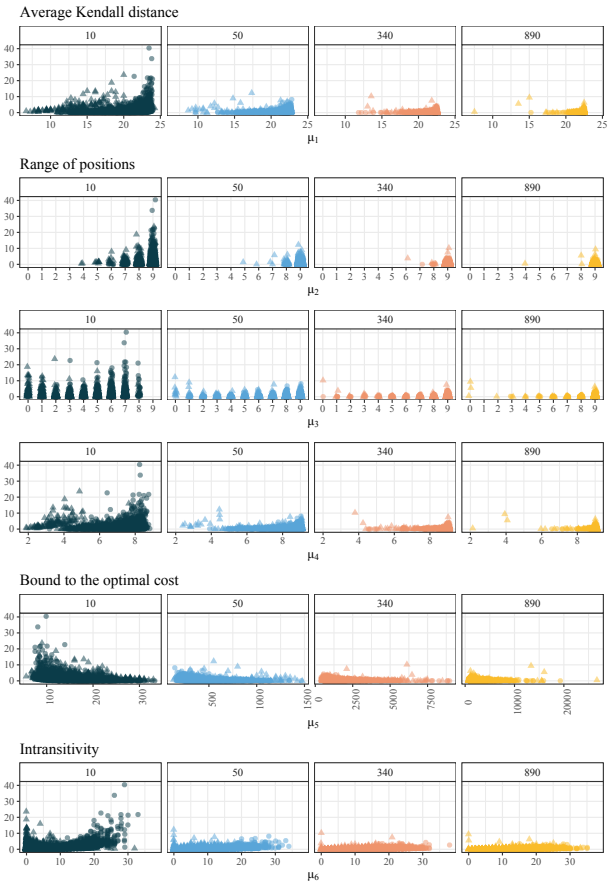


Fig. 1. Indexes already proposed in the literature in relation to the execution time.

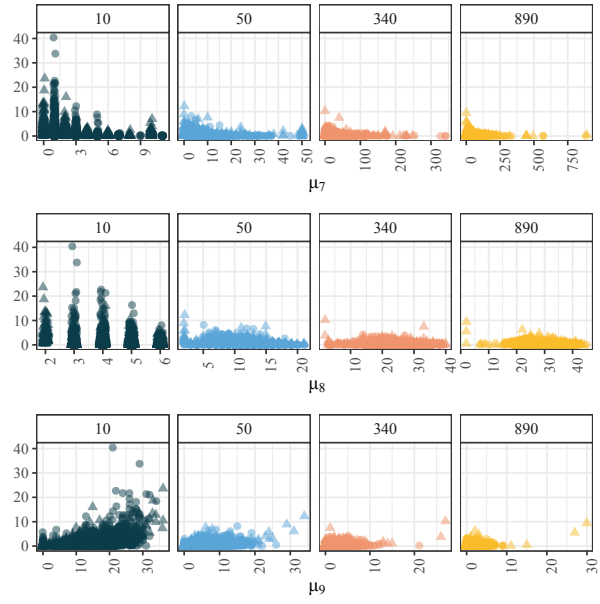


Fig. 2. Indexes based on the pairwise comparison of the votes in relation to the execution time.

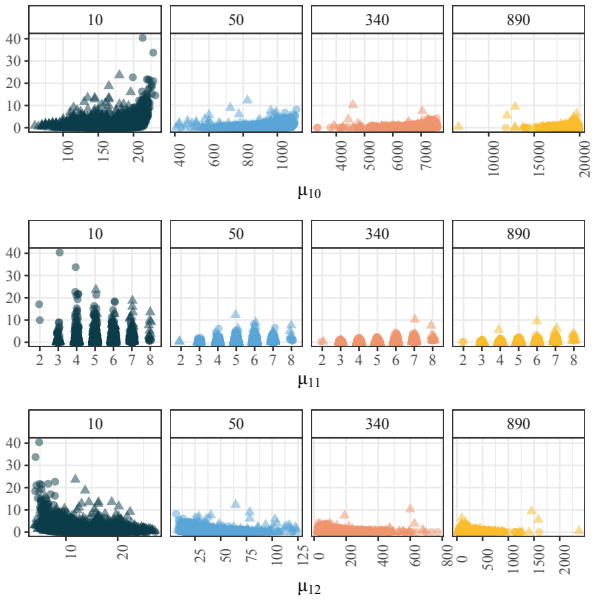


Fig. 3. Indexes based on the rowsum comparison of the votes in relation to the execution time.

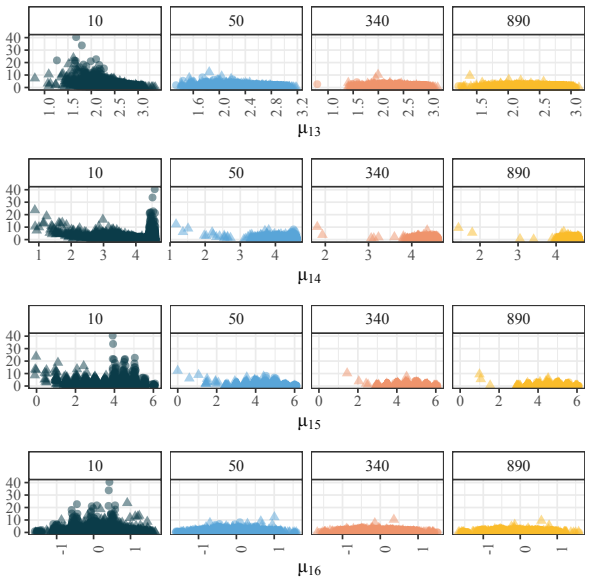


Fig. 4. Indexes based on the vector representing the Condorcet distribution in relation to the execution time.

VII. Conclusion

In this work we review the concept of difficulty in the profile of rankings in relation to the execution time that is required to achieve a consensus ranking by means of the Kemeny method. Firstly, different measures proposed in the literature regarding the agreement of the voters in the profile are gathered. Secondly, new measures are proposed in order to characterise different aspects of the profiles. Thirdly, a dataset for benchmarking containing 16000 different profiles of rankings for 10 alternatives and different numbers of voters is provided. Lastly, a study concerning the behaviour of the indexes in relation to the execution time required to find a solution for all the profiles of rankings of the generated dataset is detailed.

Although it has been seen that each index shows a behaviour that partially explains the difficulty of finding the Kemeny ranking, none of them in isolation allows us to characterise this difficulty. In particular, the indexes proposed in this paper present a different behaviour compared to those already introduced in literature (for example μ_{14}). Thus, a further study regarding the combination of some of these indexes in relation to the execution time may reveal more clear patterns. Future lines of research will include the use of these indexes in order to parametrize algorithms to ensure tractable execution times, by characterising the profiles of rankings prior to the execution according to these indexes.

Acknowledgement

This research has been supported by Grant TIN2017-87600-P from the Spanish Government and PID2019-106263RB-I00. Noelia Rico is supported by the Severo Ochoa program (PA-20-PF-BP19-167).

References

[1] Thorwald Nicolaus Tideman. Independence of clones as a criterion for voting rules. *Social Choice and Welfare*, 4(3):185–206, 1987.

[2] Markus Schulze. A new monotonic, clone-independent, reversal symmetric, and condorcet-consistent single-winner election method. *Social Choice and Welfare*, 36(2):267–303, 2011.

[3] Hobart Peyton Young. Condorcet's theory of voting. *American Political Science Review*, 82(4):1231–1244, 1988.

[4] John G. Kemeny and James Laurie Snell. *Mathematical Models in the Social Sciences*. Blaisdell book in the pure and applied sciences. Ginn, 1962.

[5] Edith Hemaspaandra, Holger Spakowski, and Jörg Vogel. The complexity of kemeny elections. *Theoretical Computer Science*, 349(3):382 – 391, 2005.

[6] J. Bartholdi, C. A. Tovey, and M. A. Trick. Voting schemes for which it can be difficult to tell who won the election. *Social Choice and Welfare*, 6(2):157–165, 1989.

[7] Felix Brandt, Vincent Conitzer, Ulle Endriss, Jérôme Lang, and Ariel D.Editors Procaccia, editors. *Handbook of Computational Social Choice*. Cambridge University Press, 2016.

[8] Ivano Azzini and Giuseppe Munda. A new approach for identifying the kemeny median ranking. *European Journal of Operational Research*, 281:388–401, 2020.

[9] Alnur Ali and Marina Meilă. Experiments with kemeny ranking: What works when? *Mathematical Social Sciences*, 64(1):28 – 40, 2012. Computational Foundations of Social Choice.

[10] Nadja Betzler, Michael R. Fellows, Jiong Guo, Rolf Niedermeier, and Frances A. Rosamond. How similarity helps to efficiently compute kemeny rankings. In *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems - Volume 1*, AAMAS '09, page 657–664, Richland, SC, 2009. International Foundation for Autonomous Agents and Multiagent Systems.

[11] Kenneth Arrow and Herve Raynaud. *Social Choice and Multicriterion Decision-Making*, volume 1. The MIT Press, 1 edition, 1986.

[12] Nadja Betzler, Michael R. Fellows, Jiong Guo, Rolf Niedermeier, and Frances A. Rosamond. Fixed-parameter algorithms for kemeny rankings. *Theoretical Computer Science*, 410(45):4554–4570, 2009. Algorithmic Aspects in Information and Management.

[13] Maurice G Kendall. A new measure of rank correlation. *Biometrika*, 30(1/2):81–93, 1938.

[14] Jean Charles Borda. *Mémoire sur les Élections au Scrutin*. Histoire de l'Académie Royale des Sciences, Paris, 1781.

XIX Conferencia de la Asociación Española para la Inteligencia Artificial

Trabajos Destacados



A Hierarchical Architecture for Recognising Intentionality in Mental Tasks on a Brain-Computer Interface

José I. Martín, Andoni Arruti, Javier Muguerza, Basilio Sierra
Computer Science Faculty, UPV/EHU
Donostia-San Sebastián, Spain

Asier Salazar-Ramírez, Raquel Martínez
Faculty of Engineering, UPV/EHU
Bilbao, Spain

***Abstract*—This article is a summary of the work published in the journal PloS ONE [1]. It presents a new proposal, based on a hierarchical structured system, to recognize intentional (IC patterns) and non-intentional mental tasks (NC patterns) on a BCI system by applying machine learning techniques to EEG signals. Our proposal achieves an average test accuracy of 66.6%, with a false positive rate of 8.2% for a modified version of BCI competition IIIa dataset, including an extra class to the original four ones regarding no movement intentionality.**

***Keywords*— Brain-computer interface (BCI); intentionality detection; electroencephalography (EEG); supervised and unsupervised learning**

I. Introduction

The work is focused on designing a hierarchical system that uses a combination of several machine learning algorithms to classify motor imaginary movements where an individual's mental process simulates a given action. Motor imagery task is one of the most studied types of BCI systems [2]. The main contribution is an approach based on a two-level hierarchical structure combining unsupervised and supervised learning algorithms to classify an individual's motor imagery mental processes to simulate given actions. The first level determines whether an activity pattern is present (IC state), or not (NC state), by applying a clustering process based on K-means algorithm and a threshold function. The second level detects which of the four mental tasks (categories of IC state: left hand, right hand, tongue and foot imaginary movements) has been produced by the user. To do so, a classifier based on Support Vector Machine paradigm is proposed.

In BCI applications, it is critical to correctly classify NC states with a low false positive rate (FPR), to avoid undesirable effects. According to the literature [3], we have selected a maximum FPR of 10%.

II. Materials and Methods

A. Data set and experimental methodology

In this work, we have used the IIIa dataset from the BCI competition III [4]. It contains data from three subjects: K3b, K6b and L1b, collected as indicated in Fig. 1 ([5]).

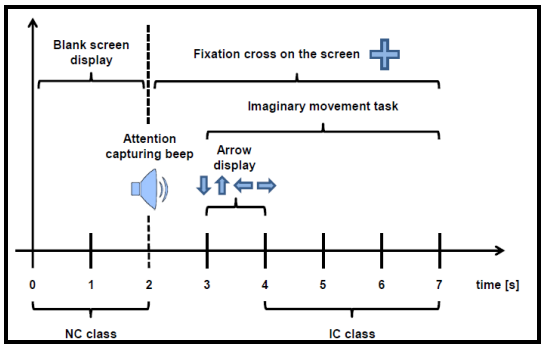


Fig. 1. Structure of an imaginary movement trials

The data was collected while the subjects were performing a cue-based (synchronized) task. To make the problem closer to a real-life operation, we also used data relating to NC states, associating the first three seconds (t = 0-3s) of each trial with no intentional activity. In this way, the classifiers must be able to differentiate between intended and unintended control brain activities. As future work, we plan to apply our approach to build an asynchronous BCI system.

The database has been pre-processed according to the work presented in [6]. For the analysis of the proposed two-level hierarchical system, we split all the data randomly into three sets: a training dataset for the first level, TrainingSet1; another training dataset for learning the second level, TrainingSet2; and a test dataset for evaluating the overall performance of the system, TestSet. For subject K3b, the first set contains 240 cases (120 correspond to IC state, namely 30 cases for each imaginary movement, and 120 to NC state); the second set contains 120 cases (120 IC state cases, namely 30 cases for each of the four classes); and the test set contains 240 cases (120 correspond to IC state and 120 to NC state). For K6b and L1b subjects, these data sets contain 160, 80 and 160 cases, respectively.

B. Previous approaches

We have tried two previous approaches using only supervised learning algorithms: (a) a one-level system composed of a classifier dealing with the five classes, and, (b) a two-level system, introducing firstly a classifier with the aim

of distinguishing between IC and NC, and presenting only the cases labelled as IC in the first level of the system to the four-class classifier of the second level. We used 11 supervised learning algorithms (Weka platform) to make these proposals. The goal of this analysis is to show the limitations that the approaches solely based on supervised learning methods have in order to keep the FPR below 10%. The best average FPR was 18.2%, and the average accuracy of 71.2%.

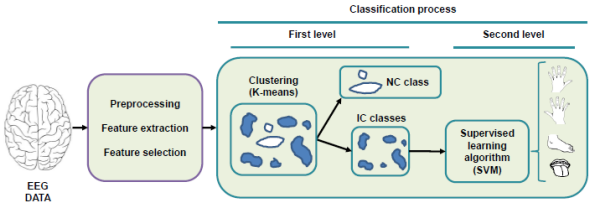


Fig. 2. Structure of the two-level hierarchical classification system

III. HIERARCHICAL CLASSIFIER

The new proposal also has a hierarchical structure. The first level determines the presence or absence of intentional activity in the EEG signal, applying clustering techniques and a threshold function. The second level determines whether the detected intentional activity is a left hand, right hand, tongue or foot imaginary movement. Fig. 2 shows a diagram that presents how the hierarchical classification process is done.

A. *First level: intentionality detection*

We used the well-known K-means algorithm (Weka implementation) with the Euclidean distance for the first level of the system. Each cluster is labelled as belonging to one of the two classes, NC or IC, taking into account that the FPR (for the NC class) must be less than the selected threshold (10%). This has been carried out by establishing a minimum percentage of IC class patterns for the generated clusters (IC-threshold) to be labelled as an IC cluster.

Therefore, for this first stage of the system, we had to select the IC-threshold and the K value for the K-means algorithm. We analyzed 12 values for the parameter K (5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55 and 60), and 4 values for the parameter IC-threshold (60%, 70%, 80% and 90%). The estimation was made by applying a 5-run, 10-fold cross-validation methodology using the TrainingSet1 dataset. Only clusters exceeding the IC-threshold were labelled as IC class. The best performance is obtained for K=35 and IC-threshold of 80%, with an accuracy of 73.5% and an FPR of 7.4%.

B. *Second level: intentional action classification*

We used supervised learning algorithms to implement this second level. The best overall classifier was the SVM algorithm, with an average accuracy of 78.2%. The best algorithm was also estimated by applying a 5-run, 10-fold cross-validation methodology, using the TrainingSet2 dataset.

To summarise, the hierarchical BCI system proposed in this work consists of a combination of a first level classifier based on a distance thresholding K-means clustering and an SVM as the second level classifier. Whereas the first level classifier is used to distinguish between IC and NC states, the second classifier is used only to determine the different classes within the clusters labelled IC. This approach maintains the FPR under de reference threshold of 10%.

C. *Experimental results (test dataset)*

Once the system has been trained, it can be used for classifying new EEG patterns, so far unknown to the system. The performance of the newly-designed system was tested using the TestSet dataset.

Analyzing the overall performance of the system in terms of classifying the five different patterns (NC state or class and 4 IC imaginary movements), the accuracy for subject K3b is 70.4%, 68.1% for subject L1b, and 59.4 for subject K6b. Taking into account all subjects, the average accuracy of the system is 66.6%, with an FPR ratio of 8.2% (Table I). These variations on the performance of the participants are produced because the three participants had been given different degrees of BCI training ([4]). Whereas K6b was a total beginner, L1b and K3b had already some experience with BCI (being K3b the one who had trained the most).

TABLE I. CLASSIFICATION PERFORMANCE OF THE PROPOSED SYSTEM WITH THE TEST DATASET

	K3b	L1b	K6b	Average
FPR (%)	7.2	8.5	9.5	**8.2**
Accuracy (%)	70.4	68.1	59.4	**66.6**

Acknowledgment

This work was partially supported by the ERDF/Spanish Ministry of Science, Innovation and Universities - National Research Agency / PhysComp project (TIN2017-85409-P), and by the Department of Education, Universities and Research of the Basque Government (ADIAN research group, grant IT980-16).

References

[1] A. Salazar-Ramírez, J.I. Martín, R. Martínez, A. Arruti, J. Muguerza, B. Sierra. “A hierarchical architecture for recognising intentionality in mental tasks on a brain-computer interface”, PloS ONE, 14(6): e0218181, 2019.

[2] G. Townsend, B. Graimann, G. Pfurtscheller. “Continuous EEG classification during motor imagery-simulation of an asynchronous BCI”, IEEE Transactions of Neural Systems and Rehabilitation Engineering, 12(2), pp. 258-265, 2004.

[3] Q. Zhao,L. Zhang, A. Cichocki.” EEG-based asynchronous BCI control of a car in 3D virtual reality environments”. Chinese Science Boulletin, 54(1), pp. 78-87, 2009.

[4] B. Blankertz, K.R. Müller, D. Krusienski, G. Schalk, J.R. Wolpaw, A. Schlögl. “The BCI2000 Competition III: Validating Alternative Approaches to Actual BCI Problems”, IEEE Transactions on Neural Systems and Rehabilitation Engineering, 14(2), pp. 153-159, 2006.

[5] A. Schlögl, F. Lee, H. Bischof, G. Pfurtscheller. “Characterization of four-class motor imagery EEG data for the BCI competition 2005”, Journal of Neural Engineering, 2(4), pp. L14-L22, 2005.

[6] O. AlZoubi, I. Koprinska,R.A. Calvo. “Classification of Brain-computer Interface Data”, 7th Australasian Data Mining Conference, vol. 87, pp. 123-132, Glenelg, South Australia, 2008.

A Self-Paced Relaxation Response Detection System Based on Galvanic Skin Response Analysis

Asier Salazar-Ramírez, Raquel Martínez, Eloy Irigoyen
Faculty of Engineering, UPV/EHU
Bilbao, Spain

José I. Martín, Andoni Arruti, Javier Muguerza
Computer Science Faculty, UPV/EHU
Donostia-San Sebastián, Spain

***Abstract*— This article is a summary of the work published in the journal IEEE Access [1]. It presents a design of a system capable of identifying automatically the Relaxation Response of a subject by analyzing a single physiological signal, the galvanic skin response. To do so, a team composed of psychologists, neurologists, and engineers designed two experiments for inducing relaxation. The F1 values obtained by the classifier scored between 0.966 and 1.000 for the data collected in both experiments.**

Keywords — Affective computing, decision trees, electrodermal activity (EDA), galvanic skin response (GSR), machine learning, relaxation response.

I. Introduction

In recent years, there has been a shift in the paradigm with regard to psychology and medicine focusing health towards physical and mental well-being and not on only the absence of disease. Relaxation techniques, among others, are one of the most commonly used methods to achieve this welfare [2]. When a person relaxes certain physiological changes take place in the organism producing what Benson [3] called a Relaxation Response (RResp) causing variations in the Autonomic Nervous System (ANS).

Several physiological signals can be useful to evaluate the state of the ANS. The Galvanic Skin Response (GSR), used to measure the amount of sweating, is sufficient to evaluate the inhibition of the Sympathetic Nervous System (SNS) produced by relaxation.

This work proposes a supervised learning algorithm, based on Decision Trees (DT), that allows physiological changes towards relaxation to be automatically detected by analyzing the GSR in 20s time windows. In addition, the algorithm not only detects these changes but also classifies them depending on their intensity.

II. Materials and Methods

This section presents the different stages of the study: the experimental process, the analysis of the data and the construction of the detection and classification algorithm itself. The sequence of these stages is depicted in Fig. 1.

A. *Experimental Setup*

This research presents two different experiments. Both have the intention of eliciting relaxation and, at the same time, recording participants' emotional states and GSR signals for later building a behavioral and physiological database.

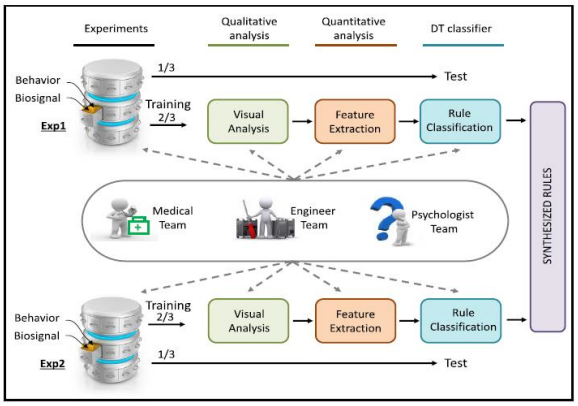


Fig. 1. Sequence of the stages until the development of the algorithm.

The first experiment (Exp1) induced stress on the participants who had been previously taken to a state of relaxation. In this experiment, in addition to the SNS activations of stress, RResps also took place in two kinds of scenarios: there is RResp produced by relaxation techniques and RResp produced by the ending of a stressful event. The population undertaking the Exp1 consisted of 166 participants (125male and 41 female) aged 19-45 years old (mean = 22.8, SD = 3.1).

Aiming to do a more thorough analysis of all possible RResps, the team designed a second experiment (Exp2) with the help of the Instituto Burmuin (a psychological medical center). In this second experiment, the psychologists induced the participants into relaxation using different techniques. It was divided into four stages, one per technique, giving a total experimental time of 12 minutes per participant. A total of 18 volunteers aged between 32 and 56 (mean = 40.22, SD=9.14) participated in the second experiment.

B. *Qualitative analysis*

As pointed out by Benson et al. [3], when an RResp takes place the SNS becomes inactive and it makes the glands stop ejecting sweat and the GSR decrease. Bearing this in mind, the researchers studied the registers and established relationships between the signals recorded in the physiological database and the behavioral database.

To do so, Fig.2 depicts the evolution of the GSR of four different subjects, two of them belonging to Exp1 (A and B) and the others belonging to two of the four stages of Exp2 (C and D). An analysis of the signals shows that the participants responded differently despite they all had done the same test.

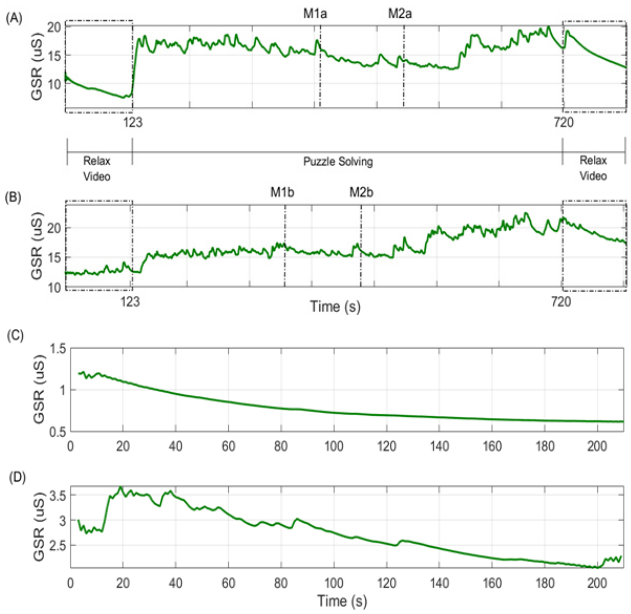


Fig. 2. Collected GSR signal registers form

After the analysis of the databases, the researchers and the experts agreed to catalogue three labels for relaxation states according to their intensity: "Low Relaxation Response"' (LRResp), "Medium Relaxation Response" (MRResp) and "High Relaxation Response" (HRResp). Besides, they also decided to include a label representing the states in which no relaxation is happening: "No Relaxation Response" (NRResp).

C. *Quantitative analysis*

It is common to approach the analysis of biosignals using sliding time window analysis [4]. In this study, the researchers decided to use a 20s window sliding every 5s. This criterion provides a sufficiently large setting to obtain information about the nervous system activation [5].

The selection of the features extracted in each window is undoubtedly one of the most significant contributions of this work since the second is a totally new feature, subject-independent and with a low computational cost. The closer the GSR is to a decreasing straight line, the greater the SNS inhibition and the higher the RResp that is taking place. Considering this, the selected features are (see Fig. 3), on the one side, the slope of the GSR (sGSR) within the window and, on the other side, the surface area (aGSR) produced by the linear regression of the GSR and the GSR itself (being both signals normalized within the analyzed window).

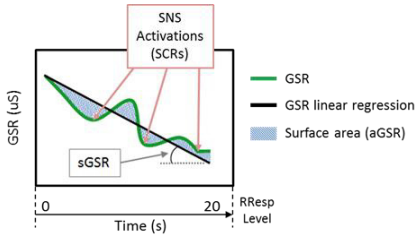


Fig. 3. The features extracted in each time window: sGSR and aGSR.

III. RESULTS

Having labeled the data from Exp1 and Exp2, 2/3 of the instances were used for building the training database and the remaining 1/3 for testing the classifiers. Using Weka platform [6], the team compared the performance indicators of 12 classification algorithms and chose to use Decision Tree (DT) for their level of success and simplicity.

The results of the system are shown in Table 4. The F1 values obtained by the decision tree classifier scored between 0.966 and 1.000 for the data collected in both experiments.

TABLE I. PERFORMANCE INDICATORS OF THE CLASSIFICATION SYSTEM.

State	Manual label	Rules					
		TP	FN	FP	P	R	F_1
Exp1	4354				0.994	0.994	0.994
NRResp	2177	2176	1	9	0.996	1.000	0.998
LRResp	349	339	10	12	0.969	0.969	0.969
MRResp	923	909	14	4	0.996	0.985	0.990
HRResp	905	905	0	0	1.000	1.000	1.000
Exp2	513				0.992	0.992	0.992
NRResp	257	257	0	2	0.992	1.000	0.996
LRResp	44	42	2	1	0.977	0.955	0.966
MRResp	87	85	2	1	0.988	0.977	0.983
HRResp	125	125	0	0	1.000	1.000	1.000

Acknowledgment

This work has been performed thanks to the ERDF/Spanish Ministry of Science, Innovation and Universities - National Research Agency / PhysComp project (TIN2017-85409-P) and the "Fundación Jesús de Gangoiti Barrera".

References

[1] Martinez, R., Salazar-Ramirez, A., Arruti, A., Irigoyen, E., Martin, J. I., & Muguerza, J.. "A self-paced relaxation response detection system based on galvanic skin response analysis". *IEEE Access*, *7*, 43730-43741, 2019.

[2] P. Francesco, M. G. Mauro, C. Gianluca, and M. Enrico, "The ef cacyof relaxation training in treating anxiety," Int. J. Behav. Consultation Therapy, vol. 5, nos. 3 4, pp. 264-269, 2009.

[3] H. Benson, J. F. Beary, and M. P. Carol, "The relaxation response" Psychiatry, vol. 37, no. 1, pp. 37 46, Feb. 1974.

[4] de Santos Sierra, A., Ávila, C. S., Casanova, J. G., & del Pozo, G. B.. "A stress-detection system based on physiological signals and fuzzy logic". IEEE Transactions on Industrial Electronics, 58(10), 4857-4865, 2011.

[5] R. Martinez, E. Irigoyen, A. Arruti, J. I. Martin, and J. Muguerza, "A realtime stress classi cation system based on arousal analysis of the nervous system by an F-state machine" Comput. Methods Programs Biomed., vol. 148, pp. 81 90, Sep. 2017.

[6] Subhani, A. R., Mumtaz, W., Saad, M. N. B. M., Kamel, N., & Malik, A. S. "Machine learning framework for the detection of mental stress at multiple levels". *IEEE Access*, *5*, 13545-13556, 2017.

Improving Bayesian Inference Efficiency for Sensory Anomaly Detection and Recovery in Mobile Robots

Manuel Castellano-Quero*, Juan-Antonio Fernández-Madrigal* and Alfonso García-Cerezo*
*Systems Engineering and Automation Department
University of Málaga, Campus de Teatinos, 29071 Málaga (Spain)
Email: {mcastellano,jafernandez,ajgarcia}@uma.es

***Abstract*—An adequate perception of the environment is essential for mobile robots to operate properly in real situations. In some cases, perception may be challenging due to the nature of the scene itself, the limited capabilities of some sensors, or even both. In this work we propose a novel methodology based on Bayesian networks that enables to capture complex relationships among sensors and also to integrate heterogeneous sources of knowledge, allowing the detection of sensory anomalies and the recovery from them. The high computational cost of inference is addressed by a new algorithm that takes advantage of our model structure. The obtained results in both simulated and real experiments with a mobile robot show that it achieves better performance and accuracy compared to other existing methods, while enhancing the robustness of the whole sensory system. All these contributions have already been published in [2].**

I. Introduction

Mobile robots are nowadays present in countless applications related to professional environments (e.g., industrial, medical, etc.) and also to domestic ones [1]. For a proper robotic operation, there are some essential tasks that must be solved (i.e., localization, mapping and navigation), all of them relying on an adequate perception of the environment. It is not sufficient in this context to extract useful information from noisy sensory data, since real environments are potentially complex and are usually governed by many different variables. Thus, a mobile robot should also be able to detect and overcome abnormal situations affecting its sensory apparatus. In robotics, this has been addressed by using different methodologies that are useful but not rigorous either in handling uncertainty or in integrating multiple sources of knowledge. To solve that, we contribute with a novel sensory anomaly detection framework, based on Bayesian networks, that is capable of handling uncertainty while capturing complex relationships among the sensors in a mobile robot. We also address the usually high computational cost of inference with Bayesian networks by introducing a new algorithm that leverages the particular structure of our model. This proposal is validated through different simulations and real experiments with a mobile robot. The obtained results show that it is possible to enhance the robustness of the sensory system while reducing the computation time of inference. Recall that this paper is a summary of the key contributions previously published in [2].

II. Representation of robotic sensory systems

The framework introduced in this work is grounded on a basic element, the so-called *Bayesian sensor*, which represents a unique aspect that could be measured by one or more sensors in the robot (e.g., a physical quantity, available information about the state of the robot or its environment, etc.). Formally, a Bayesian sensor $\mathcal{B}$ is a Bayesian network (i.e., $\mathcal{B} = (B_g, B_\theta)$ with B_g the graph and B_θ the CPTs) defined over a set of discrete random variables $\mathbf{B_z}$. At this stage of our research, all the continuous sensory quantities must be discretized before being represented. The general structure of this basic component is depicted in figure 1, and it is organized as follows:

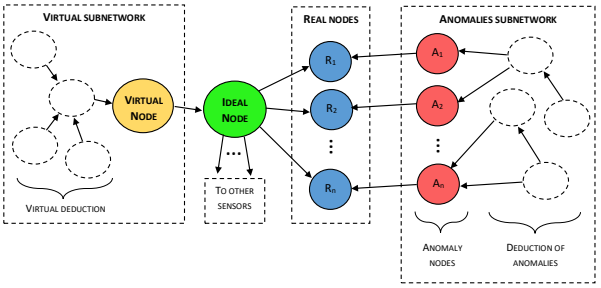
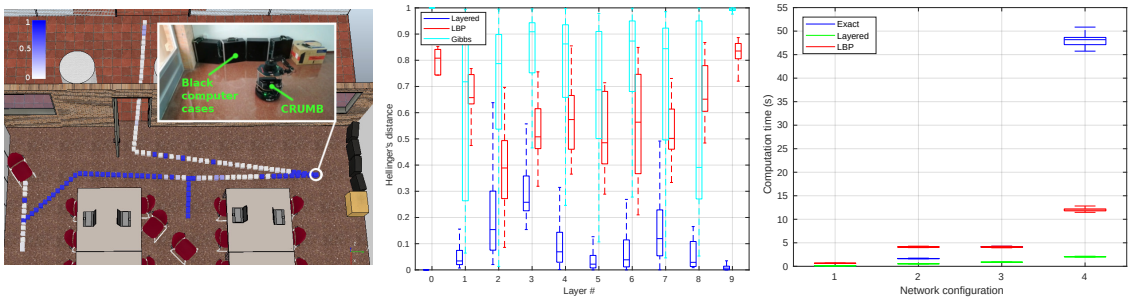


Fig. 1. Schematic representation of a generic Bayesian sensor. Nodes in dashed lines may represent either *ideal nodes* of other sensors or nodes encoding external information sources (see main text).

- **Ideal sensor node** (I): it represents the state that should be measured by the related information source(s) in the absence of abnormal conditions. This variable will be used to convey information to other sensors (as parent of other nodes).
- **Real sensor nodes** ($\mathbf{R}$): this set represents the values actually observed by the related sensory source(s).
- **Anomalies subnetwork** ($\mathcal{A}$): this is a Bayesian network, defined over variables $\mathbf{A_z}$, that serves to reason about the existence of abnormal conditions affecting sensors. They are represented by the subset of binary variables $\mathbf{A} \subseteq \mathbf{A_z}$, called *anomalies nodes*.
- **Virtual subnetwork** ($\mathcal{V}$): this is a Bayesian network, defined over variables $\mathbf{V_z}$, that is used to rebuild the datum that should be observed by the corresponding sensor by combining information from other sensors and/or external sources. The result of this reasoning is encoded in the so-called *virtual node* $V \subset \mathbf{V_z}$.

The definition of a Bayesian sensor must be completed by filling the corresponding CPTs, for instance, by using



(a) (b) (c)

Fig. 2. (a) Virtual representation of the real experimental scenario with a snapshot of the CRUMB robot during its trajectory, facing a kind of abnormal situation. The probability of such anomaly is also represented along the whole trajectory. (b) Hellinger's distance for the approximate algorithms w.r.t the exact *jointree* in the simulated experiment. (c) Computation time for the *jointree*, LBP and *layered* inference algorithms in the real experiment.

expert knowledge, statistical information about the sensor, environmental data, etc. A monolithic model of the sensory system can be then obtained by joining all the defined sensors together. For that, the existing interactions among them must be previously compiled, which can be either non-cyclic (if a sensor influences the behavior of another) or cyclic (if the influence is mutual). The latter cannot be encoded in a Bayesian network, thus, in order to break cycles and to encode all the interactions, we have devised the construction of a so-called *Dependency graph* D_G, which represents an abstraction of the monolithic model of the system.

The complete network enables to deduct the anomalies affecting a certain physical sensor, if any, and also its ideal state in a fault-free situation. These queries can be formalized as the posterior distributions $\mathrm{P}(A_i|\mathbf{E}=\mathbf{e})$ and $\mathrm{P}(I|\mathbf{E}=\mathbf{e})$, respectively, where $A_i \subset \mathbf{A}$ is an anomaly node of a certain Bayesian sensor, I is the corresponding ideal node and $\mathbf{E}$ is the set of the union of all real sensor nodes in the model, instantiated for specific values $\mathbf{e}$. Inference in such a potentially large monolithic model can be very inefficient, thus, we propose the definition of a so-called *layered* network, which splits the monolithic one into different parts named *layers* and distributes Bayesian sensors into them according to their dependencies, established in D_G. The resulting network is an approximation of the monolithic one in which all the arcs shared by different sensors in the latter are removed. We have also devised an inference algorithm for the layered network, which applies the exact *jointree* [3] to each layer in sequential order. Our algorithm works by querying the ideal variables of sensors in lower layers and encoding such distributions as the priors of the corresponding ideal variables in the higher ones. These queries will use a reduced set of evidence variables $\mathbf{E_r} \subseteq \mathbf{E}$, since all the real sensor variables in a certain layer are not available in the lower ones. Thus, the proposed algorithm produces approximate results due to two reasons: it performs inference using an approximate model and it might not consider the whole set of available evidences.

III. Implementation and Validation

We have implemented our proposal for a mobile platform, the CRUMB robot [2] (figure 2(a)), and have also validated it through simulated and real experiments in terms of error w.r.t an exact inference method (*jointree*) and computational cost. The performance has been compared to the one of other approximate algorithms, namely, the *loopy belief propagation* (LBP) and the *Gibbs sampling* [3]. One of the simulations consists in querying all the ideal variables in the model, sampling the space of the evidence set for that. The errors produced by the approximate methods in terms of the Hellinger's distance [2] are depicted in figure 2(b), where it is shown that our method always achieves the lowest median error. We have also studied the computational cost with an experiment in which our robot navigates in a real scenario as it detects and overcomes some abnormal conditions. We repeated the experiment using four different configurations of the model with increasing complexity (achieved by increasing the support cardinality of those variables representing continuous quantities). The obtained results (figure 2(c)) show that our proposal has always the lowest execution time (the Gibbs sampling method has been omitted for being extremely inefficient).

IV. Conclusions and Future Work

We introduced a methodology that enables the representation of robotic sensory systems, and proved the enhancement in robustness and efficiency. In future works, we plan to implement our proposal in a wider variety of mobile platforms and also to modify the modeling process so that it is done more autonomously.

Acknowledgment

This work has been supported by the Spanish government through the national grant FPU16/02243, by the University of Malaga through its local research program, and by the national research project RTI2018-093421-B-100.

References

[1] IFR, *Website of the IFR*, 2021. [Online]. Available: https://ifr.org/
[2] M. Castellano-Quero, J.A. Fernández-Madrigal and A. García-Cerezo, *Improving Bayesian inference efficiency for sensory anomaly detection and recovery in mobile robots*. Expert Systems With Applications, vol. 163, p. 113755, 2021.
[3] A. Darwiche, *Modeling and reasoning with Bayesian networks*. New York: Cambridge University Press, 2009, vol. 9780521884.

Template-free detection and classification of membrane-bound complexes in cryo-electron tomograms

Antonio Martinez-Sanchez
Dept. of Computer Sciences and Health Research Institute of Asturias
University of Oviedo
Oviedo, Spain
martinezsantonio@uniovi.es

***Abstract*—The application of Artificial Intelligence techniques to large amounts of data is revolutionizing many scientific fields. Concurrently, cryo-electron tomography, 3D imaging for electron microscopy at sub-nanometric resolution, has the potential to become a major technique for molecular biology and biomedicine. Here we summarize the computer method in [Nature Methods 17(2) 209-216, 2020], which is a good example of how a customized reformulation of advanced data-analysis methods applied to high-quality data enables to obtain high-impact scientific results. Specifically, by applying advanced methods for topological data analysis together with unsupervised learning, we process cellular structures previously elusive to cryo-electron tomography. Thanks to this method new transient macromolecules in their native context are observed and the synapse organization is described in the highest detail.**

Keywords—pattern recognition, unsupervised classification, image processing, cryo-electron tomography

I. Introduction

Cryo-Electron Tomography (cryo-ET) is an extension of cryo-Electron Microscopy (cryo-EM), imaging technique awarded with Nobel prize in 2017 that, among other recent achievements, has solved COVID-19 spike protein structure. With faithful sample preservation and direct imaging of fully hydrated biological material, cryo-electron tomography provides an accurate representation of the molecular architecture of cells in 3D with unprecedented resolution, thus enabling to study protein interactions *in situ* for the first time [2]. Cryo-ET and cryo-EM have the potential to become important tools for pharma industry, as well as being key to unravel the pathological mechanism behind some diseases.

Computer sciences plays an essential role in cryo-ET development. The application of advanced computational methods for image analysis and interpretation are required due to the high structural heterogeneity of subcellular architecture, low SNR, and the presence of specific anisotropic distortions.

Template Matching (TM) is the reference algorithm for macromolecule localization in cryo-tomograms. However, TM requires an initial template, so it is useless for *de novo* analysis, that is, localize and reconstruct (obtain a high-resolution 3D model) unknown macromolecular complexes. Currently, the author is contributing to prove that deep learning outperforms TM for localization tasks in 3D cellular cryo-tomograms [3]. After training, an adapted U-Net improves TM accuracy and reduces computing time. Nevertheless, this training step prevents *de novo* analysis.

In a recently published article [1], the author and others present a novel procedure for template-free picking (localization) and unsupervised classification based on structure (identification) for membrane associated macromolecules in cryo-ET images, which is implemented in an opensource software package called PySeg. Here, we briefly describe the methods included in [1], focusing on how advanced methods for topological data analysis together with unsupervised learning applied to images augment cryo-ET capabilities.

II. Methods

A. Overview

First, densities are traced and complexes detected in a comprehensive, template-free manner. Then, they are classified into classes containing structurally similar complexes, rendering them suitable for further processing. Finally, their average densities are determined to reconstruct high resolution 3D models.

B. Template-free density tracing

For an automated tracing of density in cryo-tomograms, we developed a procedure based on the discrete Morse theory, which is used to calculate topological indices of manifolds and simplicial complexes. To allow further processing, we map Morse complexes into a spatially embedded graph, such graph combines precise geometrical and external information to represent the distribution of the biological material (proteins and lipids).

Complex network of densities and a high level of noise present in cryo-tomograms results in many closely spaced local minima, causing an overly complex structure of the calculated Morse complexes. To solve this problem, we implemented a modified version of the simplification by topological persistence, which keeps just relevant densities while connectivity information is maintain by tracing accurately densities in cellular cryo-tomograms. Finally, membrane-bound particles are selected by post-processing the spatially embedded graph. To reduce complexity of any further analysis a reference system is defined for each particle associated to membrane local tangent plane and normal (see Fig 1.a).

C. Structural classification

Particles are classified (see Fig 1.b) by the Affinity Propagation (AP) clustering [4], whereby nodes (particles) exchange information between each other to reach the optimal partitioning. Thanks to AP, the approach is data driven discovering the number of classes within unbalanced datasets.

The success of a clustering procedure critically depends on the definition of the clustering distance or similarity. Here, thanks to the local reference system mentioned above, we

XXX-X-XXXX-XXXX-X/XX/$XX.00 ©20XX IEEE

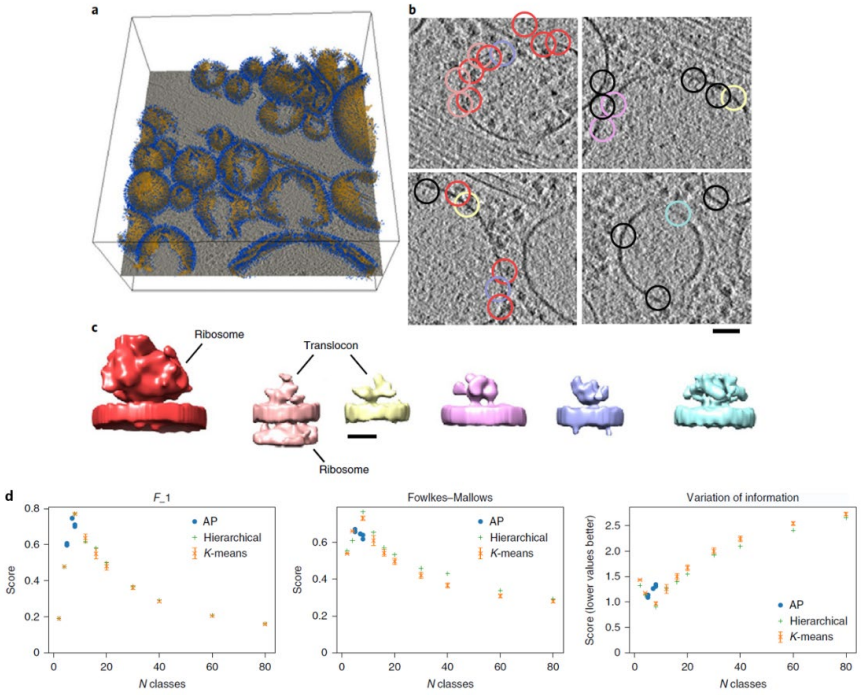


Figure 1. a) A cryo-tomogram with membranes segmented in orange and template-free particle selected as blue arrows. b) Zoom to several regions with some macromolecules localized. c) 3D density models obtained after class averaging, their resolution allows to identify some macromolecules. d) Classification approaches comparison, AP can find by itself the optimal number of classes.

compute particle rotational averages around their membrane normal vectors and define the distance between two particles as the dot product of their normalized rotational averages.

D. Structure reconstruction

Particles (sub-tomograms) for each class are aligned each other by applying rigid transformations in a procedure known as sub-tomogram averaging, this procedure provides 3D density maps virtually free of noise an distortions by averaging many copies of aligned similar structures (see Fig. 1.c). We use the gold standard procedure in the field based on Maximum A Posteriori (MAP) optimization algorithm, but we extend its performance by adding prior information from the local reference system obtained for each particle.

III. RESULTS

We evaluate our approach with synthetic data to prove that we can localize membrane-bound densities with different geometries in crowded images with very low SNR. Additionally, we compare the proposed algorithm for classification, based on AP, with those already used in the field, based on PCA for obtaining feature vectors and Hierarchical Clustering or Kmeans for clustering. This comparison shows that our approach (unsupervised) achieves a similar performance than previous literature (supervised) but without any prior knowledge about the actual number of classes embedded (see Fig. 1.d).

We also process experimental data with our approach, in cryo-tomograms containing native cellular endoplasmic reticulum we discover and visualize new spices of macromolecules elusive to any other technique. More recently, in [5], we use this method to provide the most precise description of synapse (neurons interconnexion) organization so far.

IV. DISCUSSION

The application of AI techniques in an interdisciplinary context such as Cryo-ET extends considerably the information obtained from 3D images. The tailored adaptation of advanced methods to process experimental data is key for solving long-standing pending question in molecular biology.

The success of Cryo-ET together with an improvement in acquisition hardware is increasing dramatically the amount of data. Therefore, now we plan to focus on reducing computational complexity of current data analysis bottlenecks, specifically topological simplification and unsupervised classification.

REFERENCES

Methods, figure, and results presented and discussed in this article are taken from [1].

[1] Martinez-Sanchez, A., et al. “Template-free detection and classification of membrane-bound complexes in cryo-electron tomograms,” *Nature Methods*, vol. 17(2), pp. 209-216, 2020.

[2] Lucic, V., Rigort, A. & Baumeister, W. “Cryo-electron tomography: the challenge of doing structural biology in situ,” *J. Cell Biol.*, vol. 202, pp. 407–419, 2013.

[3] Moebel, E., et al. “Deep learning improves macromolecules localization and identification in 3D cellular cryo-electron tomograms,” *bioRxiv* https://doi.org/10.1101/2020.04.15.042747 *(in revision)*.

[4] Frey, B. J. & Dueck, D. “Clustering by passing messages between data points,” *Science*, vol. 315, pp. 972–976, 2007.

[5] Martinez-Sanchez, et al. “Trans-synaptic assemblies link synaptic vesicles and neuroreceptors,” *Science Adv.*, vol. 7(10), pp. eabe6204.

Improving fake faces detection to prevent fraudulent registrations

Luis Cárabe
Departamento de Ingeniería Informática
Universidad Autónoma de Madrid
Madrid, Spain
e-mail: luiscarabe@gmail.com

Eduardo Cermeño
Research Department
Vaelsys
Madrid, Spain
e-mail: eduardo.cm@vaelsys.com

***Abstract*—Face identification is increasingly being used to register and access specific applications and online services. This opens up new possibilities for malicious attacks, such as users registering multiple times with different images or impersonating other users. Morphing is often the preferred method for these attacks as it allows the physical features of a subject to be progressively modified to resemble another subject. In [1] a method to prevent registrations using morphed images is proposed, while in [2] a new morphing scheme called stegano-morphing is tested against an improved version of the prevention system. This work highlights the improvements applied to the fake face detection system. The results suggest that it can hardly be beaten by any attack using traditional morphing techniques.**

***Index Terms*—Access control, ArcFace, biometrics, FaceNet, face identification, morphing, spoofing attack.**

I. Introduction

A morphing attack is the alteration of a subject's portrait using morphing techniques leading to his misidentification. Face recognition algorithms might be beaten or defeated by a morphing attack when the image resulting from a morphing process is not identified as the original subject. However the process of morphing might be detected using a morphing detector. This method is presented in [1] with promising results. When little morphing is applied, less than 15%, 84% of the modified images are properly detected. However, when 15% of morphing is applied the success rate drops below 80%. On the other hand, the authors show that images with more than 15% of morphing transformation are easily detected by a morphing detector. However, decreasing the amount of morphing by just 1% or 2% has a relevant impact into the confidence of the detector, that drops quickly below 90%. One of the challenges considered in [2] is to make the fake face detector system even more robust.

II. Improving fake faces detector to challenge it

The first part of the work seeks to improve the fake face detector with a new face identification algorithm. The second part focuses on testing a new attack algorithm. Experiments from [1] have shown that FaceNet, a Deep Learning based method provides significantly better results than any other kind of method. Therefore it makes sense to test other Deep Learning methods. In [2], ArcFace is used.

A. Experiments

We have created a database based on LFW [4]. As seen in [3], it is a widely used database to test state-of-the-art face recognizers. The database has 5749 subjects, but we want only the ones that have between 5 and 20 images each (both numbers included). That filters the database to 366 people with a total number of 3062 images. To determine the pairs of subjects who look more alike, we have used the Similar-looking LFW database (SLLFW) [5], which offers 3000 pairs of similar-looking faces (using the images of LFW). We have picked 25 pairs of images from it considering two factors. First, the individuals must be included in our 366 subjects database. Second, the subjects need to have more than five photos to train once the similar-looking images selected are removed from the training database. Considering all the pairs, there are 49 different images. The training database of the morphing robustness comparison and selection experiments consists of $3062 - 5 - 49 = 3008$ images of 366 subjects.

Experiments to improve the face identification module are performed by adding ArcFace to the previously tested algorithms. Then the proposed attack algorithm is tested against the best face identification method.

B. Results

Fig. 1 shows the face identification algorithms comparison of their robustness against traditional morphing. Their x-axes represent the level of morphing in the pairs. 0% morphing symbolizes the unaltered image of the first subject of the pair (original subject), 100% the second subject, and the rest of percentages the in-between morphings. Their y-axes reflect the percentage of couples who still have their original subject identified within the top for each morphing level.

By far, FaceNet and ArcFace are above all the other recognizers. They are the ones that take the longest to misidentify the original subject. We can see that the identification percentage of ArcFace is above FaceNet with low levels of morphing, but at some point, it drops below the percentage achieved with FaceNet.

Best identification scores are achieved, with 0% morphing, by ArcFace: 100%. FaceNet is in the second position achieving 84%. As ArcFace is the recognizer that achieves the highest

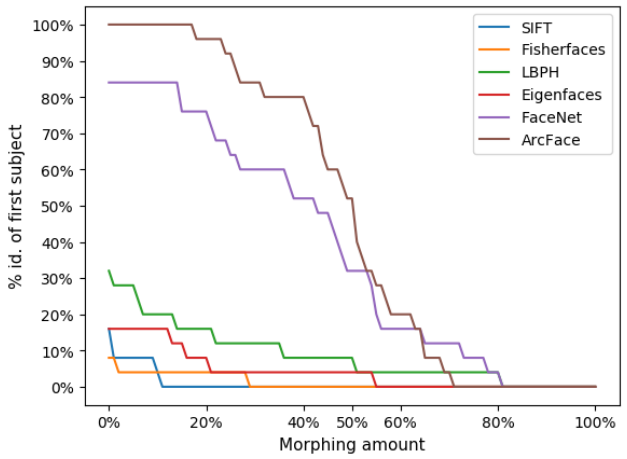


Fig. 1. Percentage of morphed images identified as the original subject for each level of morphing.

identification rate in most of the in-between morphings (in the top 1), we consider it to be the most robust recognizer.

C. Discussion

The results obtained in facial identification on the LFW database are considerably worse than those obtained in verification (except with ArcFace). This might be expected since, for identification, we work 1 vs. N ($N = 366$ in our database), and regarding verification, we work 1 vs. 1.

FaceNet and ArcFace present a good performance identifying the in-between morphed images correctly. This means that in the case of a real attack, the attacker would need to significantly alter the image to fool the recognizers.

TABLE I
ACCURACY OF FACENET (FNET) AND ARCFACE (ARCF) AT DIFFERENT PERCENTAGES OF MORPHING USING THE TRADITIONAL METHOD.

	% of morphing								
	0%	**43%**	**50%**	**51%**	**57%**	**64%**	**66%**	**71%**	**100%**
FNet	84%	*48%*	32%	32%	16%	16%	12%	12%	0%
ArcF	100%	72%	52%	*40%*	24%	16%	8%	0%	0%

The performance of the morphing detector is also remarkable. The traditional morphing is not able to reach more than 4% of complete attacks (Table II). This means that the morphing detector can detect 96% of the attacks even if the required threshold is higher than 99.93%. If it is below this value, then it detects 100% of the attacks.

TABLE II
COMPARISON OF COMPLETE ATTACKS DEPENDING ON THE MORPHING DETECTOR'S CLASSIFICATION THRESHOLD OF THE MORPHING DETECTOR USING THE TRADITIONAL (TRAD) MORPHING AND OUR PROPOSAL (PPSD). HIGHER IS BETTER.

	Threshold					
	100%	**99.46%**	**93.93%**	**1.49%**	**1.48%**	**0.09%**
Trad.	4%	4%	0%	0%	0%	0%
Ppsd.	16%	8%	8%	8%	4%	0%

III. CONCLUSIONS

Most of the existing face recognition techniques perform poorly when trying to classify images with some morphing degree. The exceptions are Deep Learning based algorithms (ArcFace) that can robustly detect 100% of the images that have less than 18% of morphing alteration.

We have presented a new way of attacking face identification systems that minimizes the chances of being detected by a morphing detector system. We have called the new method: Stegano-Morphing. The results outperform previous morphing techniques by 300% in the best case. A soft modification of 34% of the original image is enough to make the best identification algorithm misclassify almost 85% of the subjects.

However, a morphing detector algorithm is still able to catch more than 84% of the attacks. The results from this detector are awe-inspiring compared with our own human perception, which would be unable to notice any alteration on the images in many positive cases. Using machine-based auditors to verify if an image has been morphed seems necessary when trying to avoid spoofing attacks based on morphing faces.

REFERENCES

[1] L. Cárabe and E. Cermeño, "Methods to prevent registration using fake face images," in *2021 The Sixteenth International Conference on Internet Monitoring and Protection (ICIMP),* Valencia, Spain, 2021.
[2] L. Cárabe and E. Cermeño, "Stegano-morphing: Concealing attacks on face identification algorithms ," in *IEEE Access*.
[3] M. Wang and W. Deng, "Deep face recognition: A survey," 2018, *arXiv:1804.06655*. [Online]. Available: https://arxiv.org/abs/1804.06655.
[4] G. B. Huang, M. Ramesh, T. Berg and E. Learned-Miller, "Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments," University of Massachusetts, Technical Report 07-49, Oct. 2007.
[5] W. Deng, J. Hu, N. Zhang, B. Chen and J. Guo, "Fine-grained face verification: FGLFW database, baselines, and human-DCMN partnership," *Pattern Recognition,* vol. 66, pp.63–73, 2017.

Hair Segmentation and Removal in Dermoscopic Images using Deep Learning*

*Note: Work published on the IEEE Access Journal [1]

Lidia Talavera-Martínez [1,2] Pedro Bibiloni [1,2] Manuel González-Hidalgo [1,2]
[1]*SCOPIA Research Group. University of the Balearic Islands, E-07122 Palma, Spain*
[2]*Health Research Institute of the Balearic Islands (IdISBa), E-07010, Palma, Spain*
l.talavera, p.bibiloni, manuel.gonzalez@uib.es

***Abstract*—Malignant skin lesions, whose incidence rate is raising, poses a major problem for public health. When analyzing these lesions in dermoscopic images, the hairs and their shadows on the skin may occlude relevant information about the lesion at the time of diagnosis, reducing the ability of automated classification and diagnosis systems. Hence, we proposed a new approach for the task of hair removal on dermoscopic images based on deep learning techniques. To quantitatively evaluate our model, we created a dataset that contains the same images with and without hair, by simulating the presence of hair in hairless images extracted from publicly known datasets. Qualitative and quantitative results demonstrated the effectiveness of our model.**

***Index Terms*—Deep Neural Networks, Dermoscopy, Hair removal, Image Processing, Inpainting, Skin Lesion.**

I. Introduction

The diagnosis of skin lesions is mainly based on their morphological characteristics, such as an irregular shape, presence or absence of some structures, and a variety of colors. However, the existence of hair in dermoscopic images usually occludes significant patterns, and introduces uncertainty and imprecision, reducing the accuracy of the lesion assessment. Hair removal stands out as one of the key methods in the preprocessing step during the analysis of the lesions. In the literature there is extensive research within the field of computer vision that addresses the hair removal process in dermoscopic images. So far, only traditional techniques have been used for this task, even in more advanced systems where the main model uses deep learning techniques. At the work published in the IEEE Access journal [1], we presented, as far as we know, the first deep learning technique used for hair removal in dermoscopic images. Also, we introduced a novel loss function based on the combination of loss functions that focus on different aspects to complement each other towards a more robust reconstruction.

II. Methodology and Experimental Framework

a) ***Architecture structure****:* We proposed a convolutional encoder-decoder model, which is detailed in Figure 1. The choice of this architecture resided in the aim of evaluating the suitability of autoencoders to tackle this task as a denoising problem. Our hypothesis was that the network will treat the hair as noise and will be ignored, having as output the hairless skin image. In terms of its size, we believe that a rather small network was more suitable to correctly learn the task due to the relatively small amount of data we have available. Regarding its implementation, we use skip connections to enable the decoder recover image details.

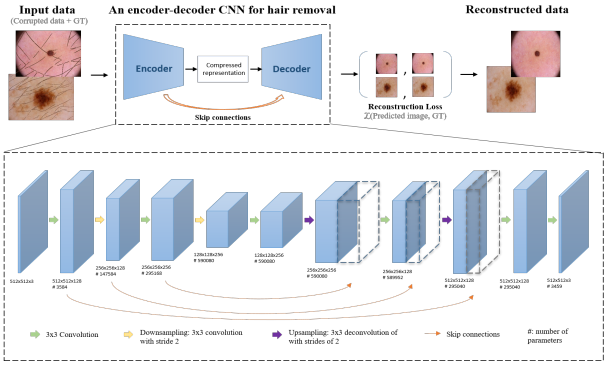


Fig. 1: Architecture of our proposed network. The pairs of reference hairless (GT) and its corrupted (hair simulated) images are passed through the encoder to extract complex features. The decoder, connected to the encoder with skip-connections, reconstructs the image.

We also presented a new reconstruction loss function, see Equation 1, based on the linear combination of loss functions that measure statistical features locally together with other per-pixel losses, which complement each other towards a more robust reconstruction. Specifically, it is based on the L_1 and L_2 distance, the total variation loss, and a loss function based on the structural similarity index metric.

$$\mathcal{L}_{\text{rec}} = \alpha L_1^{\text{foreground}} + \beta L_1^{\text{background}} + \gamma L_2^{\text{composed}} + \delta \mathcal{L}_{\text{SSIM}} + \lambda \mathcal{L}_{\text{tv}}, \quad (1)$$

where $\alpha, \beta, \gamma, \delta$ and λ are the weights of each term of the linear combination. A detailed explanation is found in [1].

b) ***Dataset****:* In order to train the model and carry out the experiments, we extended the dataset presented in [2]. It contains 618 pairs of images: images with simulated hair, used as the algorithm input, along with their corresponding "clean" version, the same image without hair. We used three different

This work was partially supported by the Spanish Grant FEDER/Ministerio de Economía, Industria y Competitividad - AEI/TIN2016-75404-P. Lidia Talavera-Martínez also benefited from the fellowship BES-2017-081264 conceded by the *Ministry of Economy, Industry and Competitiveness* under a program co-financed by the *European Social Fund*.

hair simulation methods over hairless images extracted from five publicly available dermoscopic datasets.

*c) **Experiments Results**:* We conducted a qualitative and quantitative analysis of the obtained results. In Figure 2, we can see that our proposed method attains visually appealing results on synthetic images. In addition, In Figure 3, we show its effectiveness and its generalization ability in dermoscopic images with real hair. To quantitatively assess the quality of the results of our method, we used a set of nine measures of similarity between the hairless reference images and their corresponding model output. Then, we performed a statistical test of the results, presented in Table I, to objectively study and compare the performance of our approach with that of six traditional hair removal methods. Taking into account all the considered performance measures, the proposed method outstands according to the majority of similarity measures. Finally, we conducted an ablation study of the loss terms, as well as of some aspects of the model's architecture, such as using skip connections, or convolutions/deconvolutions instead of pooling/unpooling layers. In Table II, we can see that the introduction of skip-connections does improve the results numerically, while in Figure 4, we can visualize how the network is able to create a more detailed prediction with them, especially when it comes to dermoscopic structures such as streaks or globules. However, results do not vary significantly when pooling layers are used instead of convolutional ones in our model. On the other hand, as can be seen in Table II and Figure 5, most of the performance measures and resulting images are worsen by deleting some of the terms. This is not the case when we stop computing the $L_1^{\text{background}}$. By comparing Figures 5c and 5e, we observe that when we do not use this term, the structures tend to be more blurrier. Such blurrier regions may not be penalized as much when calculating performance measures.



Fig. 2: Example of the hair removal network's output (right) on a simulated hair image (middle) vs. the reference hairless image (left).

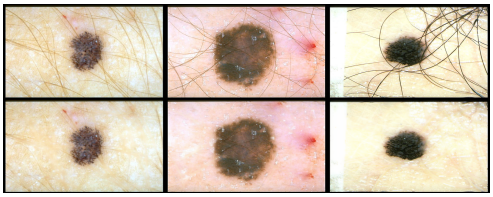


Fig. 3: Example of hair removal results (bottom) obtained by our method in dermoscopic images with real hair from EDRA2002 dataset (top).

III. Conclusions

The novel CNN-based method presented for hair removal in dermoscopic images has shown good results and its effectiveness on real hair images. Besides, the results obtained by analyzing the performance of our method and comparing it with six state-of-the-art approaches, lead to the conclusion that our method is statistically the best algorithm in eight of the nine performance measures used.

TABLE I: Mean of the similarity measures obtained to compare our method with six state of the art hair removal algorithms.

	Our method	Abbas	Bibiloni	Huang	Lee	Toossi	Xie
MSE	27.847	258.347	103.903	404.366	175.303	221.346	55.311
SSIM	0.926	0.867	0.885	0.851	0.890	0.864	0.921
PSNR	35.137	26.080	29.785	27.001	29.158	26.570	33.096
RMSE	4.790	14.226	9.207	15.485	11.032	13.287	6.318
VIF	0.525	0.526	0.499	0.402	0.531	0.509	0.592
UQI	0.997	0.991	0.996	0.990	0.995	0.992	0.997
MSSSIM	0.978	0.870	0.934	0.917	0.945	0.875	0.955
PSNR-HVS-M	36.802	25.078	29.404	26.248	28.445	25.519	33.005
PSNR-HVS	35.168	24.628	28.681	25.738	27.826	25.065	32.186

TABLE II: Mean of the similarity measures obtained on the test set images for the ablation study on our model. The second and third row correspond to the skip-connections (Skip-C) and pooling layers' ablation study. The following rows correspond to the ablation study of the loss terms.

	MSE	SSIM	PSNR	RMSE	VIF	UQI	MSSSIM	PSNR-HVS-M	PSNR-HVS
Full method	27.847	0.926	35.137	4.790	0.525	0.997	0.978	36.802	35.168
Model with,									
Pooling Layers and Skip-C	27.786	0.925	35.122	4.794	0.524	0.998	0.978	36.596	35.003
DeConv Layers and no Skip-C	32.846	0.909	33.862	5.443	0.467	0.997	0.974	35.538	33.908
Loss without,									
$L_1^{\text{foreground}}$	41.197	0.881	32.719	6.138	0.437	0.997	0.954	33.946	32.717
$L_1^{\text{background}}$	26.398	0.928	35.306	4.689	0.521	0.997	0.979	36.862	35.209
$\mathcal{L}_{\text{SSIM}}$	31.220	0.926	34.488	5.119	0.523	0.997	0.979	36.802	35.154
L_2^{composed}	30.881	0.926	34.463	5.113	0.523	0.997	0.978	36.744	35.143
$\mathcal{L}_{\text{tv}}$	30.621	0.923	34.673	5.039	0.522	0.997	0.977	36.599	34.985

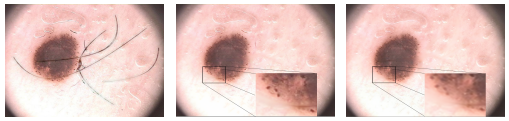


Fig. 4: Hair removal results for a sample (left), when using skip-connections (middle), and without using them (right).

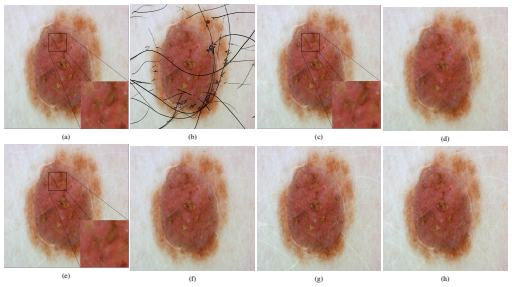


Fig. 5: (a) Original hairless image, (b) Input image with simulated hair, (c) result of our model trained with the complete loss, (d)-(h) results of our model trained by removing (d) $L_1^{\text{foreground}}$, (e) $L_1^{\text{background}}$, (f) L_2^{composed}, (g) $\mathcal{L}_{\text{SSIM}}$, and (h) $\mathcal{L}_{\text{tv}}$.

References

[1] L. Talavera-Martínez, P. Bibiloni, and M. González-Hidalgo, "Hair segmentation and removal in dermoscopic images using deep learning," *IEEE Access*, vol. 9, pp. 2694–2704, 2021.

[2] L. Talavera-Martínez, P. Bibiloni, and M. González-Hidalgo, "Comparative study of dermoscopic hair removal methods," in *ECCOMAS Thematic Conference on Computational Vision and Medical Image Processing*. Springer, 2019, pp. 12–21.

Navigating at the microscopic scale with Reinforcement Learning

Santiago Muiños-Landin,[1,4] Alexander Fischer,[1]
Viktor Holubec,[2,3] Frank Cichos[1*]

[1]Molecular Nanophotonics Group, Peter Debye Institute for Soft Matter Physics,
Universität Leipzig, 04103 Leipzig, Germany
[2]Institute for Theoretical Physics,
Universität Leipzig, 04103 Leipzig, Germany
[3]Department of Macromolecular Physics,
Faculty of Mathematics and Physics, Charles University,
18000 Prague, Czech Republic
[4]AIMEN Technology Centre, Smart Systems & Smart Manufacturing – Artificial Intelligence and Data Analytics Lab.
Pol. Ind. Cataboi, 36418 Pontevedra, Spain.

***Abstract*—Artificial microswimmers are designed to mimic the self-propulsion of microscopic living organisms to yield access to the complex behavior of active matter. Yet, as compared to their living counterparts, they have only limited ability to adapt to environmental signals or retain a physical memory to yield optimized emergent behavior. Different from macroscopic living systems and robots, both microscopic living organisms and artificial microswimmers are subject to Brownian motion which randomizes their position and propulsion direction and thus is a key feature contained in adaptation processes. Here we combine real-world artificial active particles with machine learning algorithms to explore their adaptive behavior in a noisy environment with reinforcement learning. We employ a real-time control of self-thermophoretic active particles to demonstrate the solution of a simple standard navigation problem under the inevitable influence of Brownian motion at these length scales. We show that, with external control, collective learning is possible. Concerning the learning under noise, our experiments find that noise decreases the learning speed, modifies the optimal behavior but also increases the strength of the decisions made. As a consequence of time delay in the feedback loop controlling the particles, an optimum velocity, reminiscent of optimal run-and-tumble times of bacteria, is found for the system, which is conjectured to be a universal property of systems exhibiting delayed response in a noisy environment.**

Keywords: Microrobotics, Reinforcement Learning, Self-adaptive behavior

I. Introduction

This work presented for the CAEPIA2021 conference has been recently published in Science Robotics [1]. With this work we would like to show how Machine Learning can be applied for the control of microscopic active matter, specifically artificial active matter that takes inspiration from natural organisms and their adaptability to their environment to achieve a particular goal. Information about the state of the environment is sensed, processed and encoded in biochemical processes in the organism to provide appropriate actions or properties. Such learning processes occur within the lifetime of a generation, over multiple generations, or over evolutionary relevant timescales. Leading to specific individual and collective behaviors. Swarms of fish or flocks of birds have developed collective strategies adapted to the existence of predators, or have learnt how to use convective air flows [2] and sperms have evolved complex swimming patterns to explore chemical gradients in chemotaxis .

Inspired by such optimization processes, learning strategies reducing the complexity of the physical and chemical processes in living matter to a mathematical procedure have been developed. Many of these learning strategies have been implemented into robotic systems . One particular framework is reinforcement learning (RL) in which an agent gains experience by interacting with its environment [3]. The value of his experience relates to rewards connected to the states the agent can occupy. The learning process then maximizes the cumulative reward for a chain of actions to obtain the so called *policy*. This policy advises the agent which action to take. Recent computational studies, for example, reveal that reinforcement learning can provide optimal strategies for the navigation of active particles through flows [4], the swarming of robots, or the development of collective motion [5].

Artificial microswimmers are a class of materials that integrate the key functionality of persistent directed motion, common to their biological counterparts, into a user-designed microscopic object. Their motility has already revealed insights into a number of fundamental processes, including collective phenomena [6], and they are explored for drug delivery [7] and environmental purposes [8]. Yet, the integration of energy supply, sensing, signal processing, memory, and propulsion into a micrometer-sized artificial swimmer still remains a technological challenge .

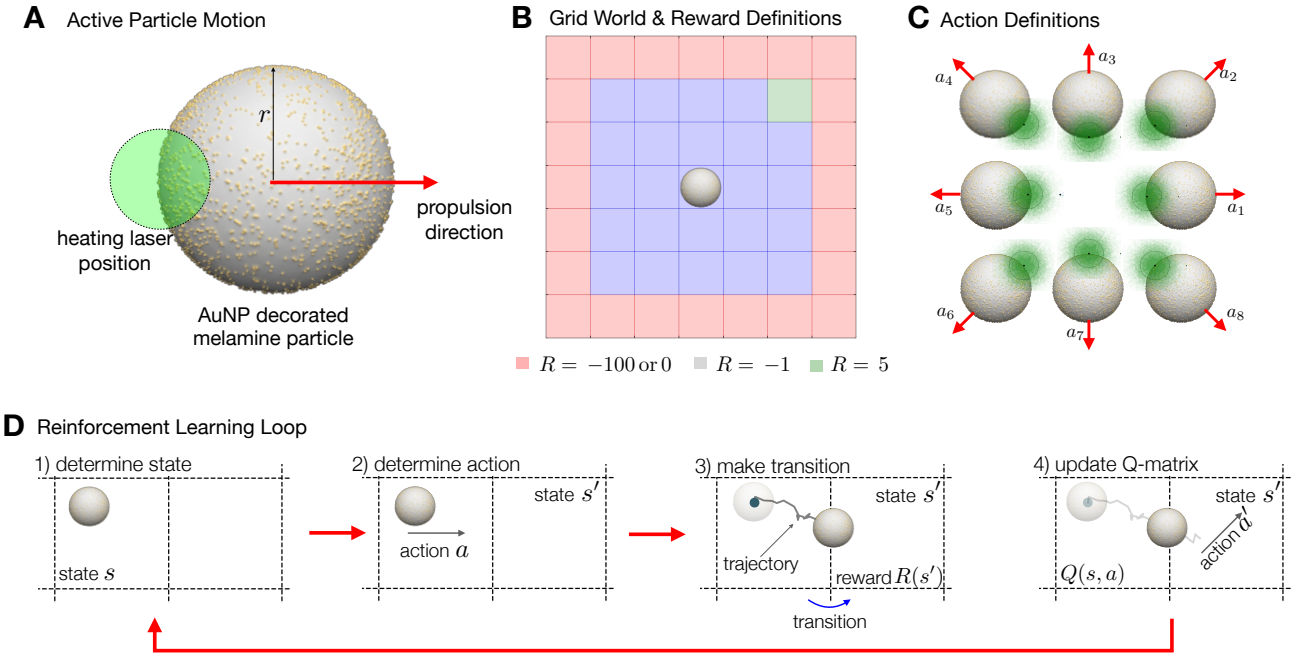


Fig. 1. **Swimmer, States and Actions. A)** Sketch of the self-thermophoretic symmetric microswimmer. The used particles have an average radius of $r = 1.09$ μm and are covered on 30 % of their surface with gold nanoparticles of about 10 nm diameter. A heating laser illuminates the colloid asymmetrically (at a distance d from the center) and the swimmer acquires a well-defined thermophoretic velocity v. **B)** The grid world contains 25 inner states (blue) with one goal at the top right corner (green). A set of 24 boundary states (red) is defined for the study of the noise influence. **C)** In each of the states we consider 8 possible actions in which the particle is thermophoretically propelled along the indicated directions by positioning the laser focus accordingly. **D)** The reinforcement learning loop starts with measuring the position of the active particle and determining the state. For this state, a specific action is determined with the ϵ greedy procedure (see Supplementary Information for details). Afterwards, a transition is made, the new state is determined, and a reward for the transition is given. Based on this reward, the Q-matrix is updated and the procedure starts from step 1 until an episode ends by reaching the goal or exiting the grid-world to a boundary state.

II. Work

Hence, external control strategies have been applied to introduce sensing and signal processing, yet only schemes with rigid rules are developed . Here, we use algorithms of reinforcement learning with external control into the motion of artificial microswimmers in an fluid. While the learning algorithm is running on a computer, we control a real agent acting in a real world subjected to thermal fluctuations, hydrodynamic and steric interactions, and many other influences. In this way it is possible to include real-world objects in a simulation, which will help to close the so-called reality gap,i.e. the difference of pure *in-silico* learning and real-world in machine learning even at microscopic length scales .

III. Conclusions

Our experimental investigation thus goes beyond previous purely computational studies [4]. It allows us to observe the whole learning process optimizing parameters, which are not accessible in studies of biological species, and to identify the most important ingredients of the real dynamics and set up new realistic but relatively simple models based on this information. It also provides a first glimpse on the challenges of reinforcement learning for objects at those length scales for future developments.

References

[1] S. Muiños-Landin, A. Fischer, V. Holubec, and F. Cichos, "Reinforcement learning with artificial microswimmers," *Science Robotics*, vol. 6, no. 52, 2021.

[2] G. Reddy, A. Celani, T. J. Sejnowski, and M. Vergassola, "Learning to soar in turbulent environments," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 113, no. 33, pp. E4877–E4884, 2016.

[3] R. S. Sutton and A. G. Barto, "Reinforcement Learning: An Introduction," *MIT Press, Cambridge*, 1998.

[4] K. Gustavsson, L. Biferale, A. Celani, and S. Colabrese, "Finding efficient swimming strategies in a three-dimensional chaotic flow by reinforcement learning," *European Physical Journal E*, vol. 40, no. 12, 2017.

[5] K. Ried, T. Müller, and H. J. Briegel, "Modelling collective motion based on the principle of agency: General framework and the case of marching locusts," *PLoS ONE*, vol. 14, no. 2, pp. 1–20, 2019.

[6] J. Palacci, S. Sacanna, A. P. Steinberg, D. J. Pine, and P. M. Chaikin, "Living Crystals of Light-Activated Colloidal Surfers," *Science*, vol. 339, pp. 936–940, 2 2013.

[7] Z. Wu, J. Troll, H. Jeong, Q. Wei, M. Stang, F. Ziemssen, Z. Wang, M. Dong, S. Schnichels, T. Qiu, and P. Fischer, "A swarm of slippery micropropellers penetrates the vitreous body of the eye," *Science Advances*, vol. 4, no. 11, 2018.

[8] M. Safdar, J. Simmchen, and J. Jänis, "Light-driven micro- and nanomotors for environmental remediation," *Environmental Science: Nano*, vol. 4, no. 8, pp. 1602–1616, 2017.

Online Learning for Probabilistic Load Forecasting

Verónica Álvarez*, Santiago Mazuelas†, and José A. Lozano‡
BCAM-Basque Center for Applied Mathematics, Bilbao, Spain
Email: *valvarez@bcamath.org, †smazuelas@bcamath.org, ‡jlozano@bcamath.org

***Abstract*—Load forecasting is crucial for multiple energy management tasks. Conventional load forecasting techniques obtain single-value load forecasts by exploiting consumption patterns of past load demand. However, such techniques cannot assess intrinsic uncertainties in load demand, and cannot capture dynamic changes in consumption patterns. To address these problems, this paper presents a method for probabilistic load forecasting based on the online learning of hidden Markov models. We develop adaptive online learning techniques that update model parameters recursively, and sequential prediction techniques that obtain probabilistic forecasts using the most recent parameters. The performance of the method is evaluated using datasets corresponding with regions that have different sizes and display assorted time-varying consumption patterns. The results show that the proposed method can significantly improve the performance of existing techniques for a wide range of scenarios.**

***Index Terms*—Load forecasting, probabilistic load forecasting, online learning, hidden Markov model.**

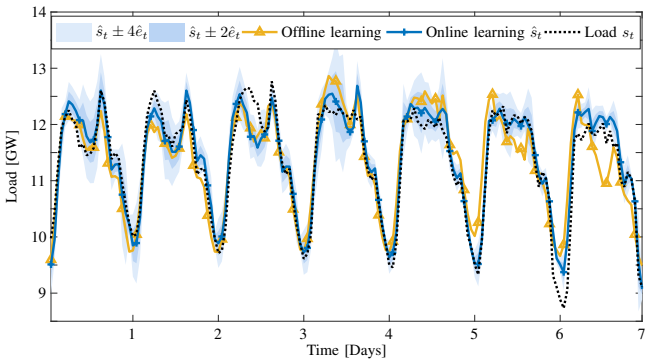


Fig. 1. The model obtained by offline learning methods cannot accommodate to dynamic changes in consumption patterns, while the models obtained by online learning methods can adapt to such changes. In addition, methods that probabilistic forecasts can assess uncertainties in load demand.

I. Introduction

LOAD FORECASTING is crucial for multiple energy management tasks. The importance of load forecasting is growing significantly in recent years due to the increasing development of power systems and smart grids. In addition, accurate load forecasting has a beneficial impact in environment and economy by reducing energy waste and purchase. Forecasting methods are enabled by exploiting consumption patterns related to multiple factors such as past loads, hours of day, days of week, holidays, and temperatures.

Accurate forecasting is hindered by intrinsic uncertainties in load demand and dynamic changes in consumption patterns. Uncertainties in load demand cannot be assessed by methods that obtain single-value forecasts, and dynamic changes in consumption patterns cannot be captured by methods based on offline learning of static models. On the other hand, probabilistic forecasts can evaluate load uncertainty and are essential for optimal stochastic decision making while online learning is necessary to harness dynamic changes in consumption patterns [1] (see Fig. 1).

Most conventional techniques for load forecasting obtain single-value forecasts based on offline learning such as seasonal autoregressive integrated moving average (SARIMA) [2] and deep residual networks (DRN) [3]. Existing techniques that obtain probabilistic forecasts are based on offline learning, while those based on online learning obtain single-value forecasts such as Gaussian process (GP) [4] and quantile regression (QR) [5]. Current online learning methods adapt to dynamic changes in consumption patterns by adjusting offline learning algorithms such as [6] that updates the smoothing functions in additive models.

In this paper, we summarize the techniques presented in [1]. Such paper presents techniques for adaptive probabilistic load forecasting (APLF) that can harness changes in consumption patterns and assess load uncertainties. Specifically, we model the data using hidden Markov models (HMMs) then, we develop online learning techniques for APLF that update HMM parameters recursively and sequential prediction techniques for APLF that obtain probabilistic forecasts using the most recent HMM parameters. In addition, we quantify the performance improvement of APLF in comparison with the state-of-the-art under multiple challenging scenarios.

II. Problem Formulation and Theoretical Results

Load forecasting methods estimate future loads given past loads and factors that affect load demand such as hours of day, days of week, and weather forecasts. Forecasting techniques determine a prediction function that assigns instance vectors $\mathbf{x}$ composed by past loads and observations related to future loads (e.g., weather forecasts) to target vectors $\mathbf{y}$ composed by L future loads. Furthermore, each time t is categorized by a calendar variable $c(t) \in \{1, 2, ..., C\}$ that describes time factors affecting load demand such as hour of day and day of week.

In the following, we summarize the theoretical guarantees for APLF learning and prediction presented in [1]. We model the relationship between the loads $\{s_t\}_{t\geq 1}$ and observations $\{\mathbf{r}_t\}_{t\geq 1}$ using non-homogeneous HMMs [7] whose models

change in time together with changes in consumption patterns. Such models allow to predict hidden states from past states and observations, and are determined by the conditional distribution $p\left(s_t|s_{t-1}\right)$ that represents the relationship between two following loads, and the conditional distribution $p\left(\mathbf{r}_t|s_t\right)$ that represents the relationship between each load and observations vector. These conditional distributions $p\left(s_t|s_{t-1}\right)$ and $p\left(\mathbf{r}_t|s_t\right)$ are modeled using Gaussian distributions with mean $\mathbf{u}_s^{\mathrm{T}}\boldsymbol{\eta}_{s,c}$ and $\mathbf{u}_r^{\mathrm{T}}\boldsymbol{\eta}_{r,c}$ and standard deviations $\sigma_{s,c}$ and $\sigma_{r,c}$, respectively. Parameters $\boldsymbol{\eta}_{s,c}, \boldsymbol{\eta}_{r,c}, \sigma_{s,c}$, and $\sigma_{r,c}$ are different for each calendar type $c(t) \in \{1, 2, ..., C\}$ and $\mathbf{u}_s = [1, s_{t-1}]^{\mathrm{T}}$ and $\mathbf{u}_r = u_r(\mathbf{r}_t)$ are known feature vectors.

Mean parameters $\boldsymbol{\eta}$ and standard deviation parameters σ can be obtained by maximizing the weighted log-likelihood of all the loads obtained at times with the same calendar type. Specifically, if $s_{t_1}, s_{t_2}, ..., s_{t_n}$ are loads obtained at times with calendar type $c \in \{1, 2, ..., C\}$, and $\mathbf{u}_{t_1}, \mathbf{u}_{t_2}, ..., \mathbf{u}_{t_n}$ are the corresponding feature vectors for parameters of each conditional distribution, the exponentially weighted log-likelihood of loads up to time t_i, for $i = 1, 2, ..., n$, can be approximatelly solved using parameters given by

$$\boldsymbol{\eta}_i = \boldsymbol{\eta}_{i-1} + \frac{\mathbf{P}_{i-1}\mathbf{u}_{t_i}}{\lambda + \mathbf{u}_{t_i}^{\mathrm{T}}\mathbf{P}_{i-1}\mathbf{u}_{t_i}}\left(s_{t_i} - \mathbf{u}_{t_i}^{\mathrm{T}}\boldsymbol{\eta}_{i-1}\right)$$

$$\sigma_i = \sqrt{\sigma_{i-1}^2 - \frac{1}{\gamma_i}\left(\sigma_{i-1}^2 - \frac{\lambda\left(s_{t_i} - \mathbf{u}_{t_i}^{\mathrm{T}}\boldsymbol{\eta}_{i-1}\right)^2}{\lambda + \mathbf{u}_{t_i}^{\mathrm{T}}\mathbf{P}_{i-1}\mathbf{u}_{t_i}}\right)}$$

with

$$\mathbf{P}_i = \frac{1}{\lambda}\left(\mathbf{P}_{i-1} - \frac{\mathbf{P}_{i-1}\mathbf{u}_{t_i}\mathbf{u}_{t_i}^{\mathrm{T}}\mathbf{P}_{i-1}}{\lambda + \mathbf{u}_{t_i}^{\mathrm{T}}\mathbf{P}_{i-1}\mathbf{u}_{t_i}}\right), \gamma_i = 1 + \lambda\gamma_{i-1}$$

for $\boldsymbol{\eta}_0 = \mathbf{0}_K$, any σ_0, $\mathbf{P}_0 = \mathbf{I}_K$, $\gamma_0 = 0$, and $\lambda \in (0, 1)$. These recursions allow to update mean and standard deviation parameters ($\boldsymbol{\eta}_{s,c}, \boldsymbol{\eta}_{r,c}, \sigma_{s,c}$, and $\sigma_{r,c}$) every time new loads and observations are obtained corresponding with calendar type c [1].

The HMM characterized by parameters described above allows to recursively obtain probabilistic load forecasts $N\left(s_{t+i}; \hat{s}_{t+i}, \hat{e}_{t+i}\right)$ where means $\hat{s}_{t+i}$ and standard deviations $\hat{e}_{t+i}$, for $i = 1, 2, ..., L$, can be computed by the following recursions

$$c = c\left(t + i\right), \hat{\mathbf{u}}_s = [1, \hat{s}_{t+i-1}]^{\mathrm{T}}, \mathbf{u}_r = u_r\left(\mathbf{r}_{t+i}\right)$$

$$\hat{s}_{t+i} = \frac{\hat{\mathbf{u}}_s^{\mathrm{T}}\boldsymbol{\eta}_{s,c}\sigma_{r,c}^2 + \mathbf{u}_r^{\mathrm{T}}\boldsymbol{\eta}_{r,c}\left(\sigma_{s,c}^2 + \left(\mathbf{v}^{\mathrm{T}}\boldsymbol{\eta}_{s,c}\right)^2\hat{e}_{t+i-1}^2\right)}{\sigma_{r,c}^2 + \sigma_{s,c}^2 + \left(\mathbf{v}^{\mathrm{T}}\boldsymbol{\eta}_{s,c}\right)^2\hat{e}_{t+i-1}^2}$$

$$\hat{e}_{t+i} = \sqrt{\frac{\sigma_{r,c}^2\left(\sigma_{s,c}^2 + \left(\mathbf{v}^{\mathrm{T}}\boldsymbol{\eta}_{s,c}\right)^2\hat{e}_{t+i-1}^2\right)}{\sigma_{r,c}^2 + \sigma_{s,c}^2 + \left(\mathbf{v}^{\mathrm{T}}\boldsymbol{\eta}_{s,c}\right)^2\hat{e}_{t+i-1}^2}}$$

for $\mathbf{v} = [0, 1]^{\mathrm{T}}$, $\hat{s}_t = s_t$, $\hat{e}_t = 0$, and $i = 1, 2, ..., L$. These recursions allow to obtain load forecasts $\hat{s}_t$ and estimates of their accuracy $\hat{e}_t$ using the most recent parameters every time new instance vectors $\mathbf{x}$ are obtained [1].

III. Numerical results

APLF method is compared with 5 state-of-the-art techniques using 3 publicly available datasets. The assorted datasets correspond with regions that have different sizes (Belgium, Dayton, and 100 buildings in New South Wales) and display different consumption patterns that change over time. The root mean square error (RMSE) and the mean average percentage error (MAPE) assessing overall prediction errors are given in Table I and Pinball loss and ECE assessing probabilistic forecasts are given in Table II. Such tables show that APLF method achieves remarkable results both in terms of single-value and probabilistic forecasts, and adapts to different consumption patterns in every region studied. In [1] we provide additional results and implementation details.

TABLE I
RMSE AND MAPE OF PREDICTION ERRORS.

Method	Belgium		Dayton		100 builds.	
	[GW]	[%]	[GW]	[%]	[kW]	[%]
SARIMA	0.81	5.5	0.22	7.7	0.07	16.5
QR	1.05	9.2	0.20	7.1	0.12	26.8
GP	**0.52**	**4.0**	0.19	6.3	**0.05**	**11.2**
DRN	1.74	13.0	0.31	11.3	0.07	14.7
AFF	0.95	6.7	0.26	9.6	0.07	14.6
APLF	**0.33**	**2.3**	**0.16**	**5.5**	**0.05**	**11.0**

TABLE II
PINBALL LOSS AND ECE.

Method	Belgium		Dayton		100 builds.	
	[GW]	ECE	[GW]	ECE	[kW]	ECE
QR	0.34	**0.08**	0.09	**0.06**	0.03	**0.08**
GP	**0.14**	0.19	**0.04**	0.12	**0.01**	**0.03**
APLF	**0.11**	**0.07**	**0.04**	**0.05**	**0.01**	**0.08**

IV. Conclusion

This paper proposes techniques for adaptive probabilistic load forecasting (APLF) that can adapt to changes in consumption patterns and assess uncertainties in load demand. We developed online learning techniques that update model parameters using a recursive algorithm, and prediction techniques that obtain probabilistic forecasts using the most recent HMM parameters. The experimental results show the performance improvement of APLF method in terms of prediction errors and probabilistic forecasts in regions that have different sizes.

References

[1] V. Alvarez, S. Mazuelas, and J. A. Lozano, "Probabilistic load forecasting based on adaptive online learning," *IEEE Transactions on Power Systems*, vol. 36, no. 4, pp. 3668 – 3680, Jul. 2021.

[2] N. Mohamed, M. H. Ahmad, and Z. Ismail, "Short term load forecasting using double seasonal ARIMA model," in *Proceedings of the Regional Conference on Statistical Sciences*, vol. 10, 2010, pp. 57–73.

[3] K. Chen, K. Chen, Q. Wang, Z. He, J. Hu, and J. He, "Short-term load forecasting with deep residual networks," *IEEE Transactions on Smart Grid*, vol. 10, no. 4, pp. 3943–3952, 2018.

[4] J. Lourenço and P. Santos, "Short term load forecasting using Gaussian process models," *Proceedings of Instituto de Engenharia de Sistemas e Computadores de Coimbra*, 2010.

[5] Y. Wang, Q. Chen, N. Zhang, and Y. Wang, "Conditional residual modeling for probabilistic load forecasting," *IEEE Transactions on Power Systems*, vol. 33, no. 6, pp. 7327–7330, 2018.

[6] A. Ba, M. Sinn, Y. Goude, and P. Pompey, "Adaptive learning of smoothing functions: Application to electricity load forecasting," in *Advances in Neural Information Processing Systems*, 2012, pp. 2510–2518.

[7] S. Mazuelas, Y. Shen, and M. Z. Win, "Belief condensation filtering," *IEEE Transactions on Signal Processing*, vol. 61, no. 18, pp. 4403–4415, 2013.

CCE: An ensemble architecture based on coupled ANN for solving multiclass problems

M. Paz Sesmero
Computer Science Department
Universidad Carlos III de Madrid
Madrid, Spain
msesmero@inf.uc3m.es

Juan M. Alonso-Weber
Computer Science Department
Universidad Carlos III de Madrid
Madrid, Spain
jmaw@ia.uc3m.es

Araceli Sanchis
Computer Science Department
Universidad Carlos III de Madrid
Madrid, Spain
masm@inf.uc3m.es

***Abstract*—This is a summary of our recent work published in Information Fusion** [1]**, which extends the experimental work started in two previous works** [2][3]**.**

Keywords: —Ensemble of classifers, Multiclass_classification tasks, Artificial Neural Networks, Diversity

I. SUMMARY

One of the main challenges in the field of the ensembles of classifiers is to achieve a balance between the error rate of the base learners and their diversity. To analyze the influence of both, diversity and accuracy of the base learners on the accuracy of the ensemble, we have designed two different ensemble architectures that are complementary. The first proposal, named *BCE* − Binary-Complementary Ensemble – and described in [2], builds base learners that are highly accurate but not very diverse. To achieve this goal, the members of the ensemble are implemented with two coupled classifiers: a binary classifier and a multiclass classifier. The binary classifier is trained to distinguish whether an example belongs to a certain class. The second classifier, named complementary classifier, is a multi-class classifier with *k-1* outputs (where *k* is the number of classes) whose goal is to label the instances that have been labelled as negative by the corresponding binary classifier. To promote diversity among the classifiers and to improve the learning time in terms of both accuracy and computational cost, each one of the classifiers that compose a base learner, is trained using a specific feature subset. The second architecture, named *CCE* − Complementary- Complementary Ensemble – and presented in [1], generates base learners that are relatively accurate but highly diverse. To accomplish this requirement, the initial learning task is transformed into a pool of new pairwise disjoint subproblems. So, an initial classification problem with instances belonging to one of *k* classes is transformed in a pool of pairwise disjoint subproblems. One of these subproblems works with the instances that belong to *j* classes ($2 \leq j \leq k\text{-}2$) and the other one with the examples that belong to the (k-j) remaining classes. Both architectures have been conceived with the goal to obtain classifier systems that are both accurate, and efficient in terms of training time.

To obtain a global vision of the behavior of both, *BCE* and *CCE*, the following aspects were analyzed: ensemble performance (classification accuracy, F1-score, and training time), diversity of the base learners, and the tolerance in the presence of examples with incorrect class labels. Moreover, to test how well these systems work, their performance were compared to that obtained by a single ANN, the OAA architecture [4], *Bagging* [5], ECOC [6], *Boosting* with resampling [7] and the OAO architecture [8].

For all the classification models, the ANN's used were one-hidden-layer Multilayer Perceptron (MLP) trained with the Back-Propagation algorithm. For each data set, the ANN architecture and topology were fixed with the objective of achieving a single MLP with a good generalization capacity.

To evaluate the performance of the different classification models, we accomplished 5 replications of a 2-fold stratified cross validation and quantify both, accuracy, and F_1-score. To statistically compare the accuracy of *CCE* with that obtained by BCE and the baseline classification models in an specific domain we applied the F-test [9]. In addition, to analyze the performance of CCE over the 20 datasets and compare it with the global performance of BCE and the baseline classification models, we applied the Wilcoxon Signed-Ranks test [10] over both, accuracy and F1-score values. Finally, to estimate the effectiveness of the proposed models over the other systems, an analysis of the training time of each system was performed.

Secondly, to study the relationship between the diversity of the base learners and the ensemble accuracy, some well-known measures of diversity were computed: The Q statistic [11], the correlation coefficient (ρ) [11], the kappa statistic (κ) [12], and the fail/non-fail disagreement measure [13]. Then, we measured the correlation between the diversity and the gain (ensemble accuracy minus mean accuracy of its base learners) of the ensemble. To check whether the gain of the different ensembles is a consequence of the diversity of their base learners, we computed the correlation between both measures using the Spearman's correlation coefficient (RCC) [13].

Lastly, to study the noise resilience of *BCE* and *CCE* we analyzed their performance in the presence of four labeling noise levels: 10%, 15%. 20% and 25%. Again, the accuracy and F1-score values of both systems were compared with those obtained by the reference classification models.

The results of the experiments carried out in [1] indicated that, in the absence of labeling noise, *CCE* significantly

This work has been supported by the Spanish Government under projects TRA2016-78886-C3-1-R, PID2019-104793RB-C31, RTI2018-096036-B-C22, PEAVAUTO-CM-UC3M and by the Region of Madrid's Excellence Program (EPUC3M17).

outperforms OAA and ANN in 11 data sets, ECOC in 10 data sets, *BCE* in 8 data sets, and *Bagging* and OAO in 2. *Boosting* outperforms CCE with statistical significance in 4 domains. For the Mnist data set, the mean accuracy rate achieved by *CCE* was statistically better than those obtained by the rest of implemented models.

Regarding to the training time, we experimentally proved that *CCE* is more efficient than most standard classification models. Only OAO, and sometimes *BCE*, require a lower training time than *CCE*.

Considering simultaneously the accuracy and the training time, we could conclude that *CCE* outperforms the other analyzed systems: *CCE* is more accurate than those systems that require less training time (OAO and *BCE*) and it requires a much shorter training time than the system with a comparable accuracy rate (*Boosting*).

Computing the correlation between the diversity and the gain of *CCE* we could check that, in effect, the members of *CCE* are relatively accurate but quite diverse and that the relationship between diversity and the gain is statistically significant.

Finally, to analyze the *CCE* performance in the presence of labeling noise, the experimental results shown that *CCE* is as accurate as *Bagging* and less prone to noise than the other implemented classification systems. This resilience, together with the shorter training time highlights *CCE* as a very good option for classification in multiclass problems.

It is worth mentioning that when the Wilcoxon signed-ranks test was used to compare the global performance of *CCE* with the baseline classifiers, the results revealed that, in the absence of labeling noise, *CCE* is statistically better than ANN, OAA, ECOC, *Bagging* OAO and *BCE* and statistically equivalent to *Boosting*. On the other hand, in presence of labeling noise, the results indicated that *CCE* is statistically better than *BCE*, ANN, OAA, ECOC, OAO and *Boosting* and statistically equivalent to *Bagging*.

From all the performed experiments, we could determine that, in multiclass problems, *CCE* offers a high correctly classified instance rate, is tolerant to labeling noise and is computational efficient. Moreover, considering these three parameters simultaneously, we could conclude that *CCE* outperforms many other classifier systems.

References

[1] M. P. Sesmero, J. M. Alonso-Weber, and A. Sanchis, "CCE: An ensemble architecture based on coupled ANN for solving multiclass problems," *Inf. Fusion*, vol. 58, pp. 132–152, Jun. 2020, doi: 10.1016/J.INFFUS.2019.12.015.

[2] M. P. Sesmero, J. M. Alonso-Weber, G. Gutierrez, A. Ledezma, and A. Sanchis, "An ensemble approach of dual base learners for multi-class classification problems," *Inf. Fusion*, vol. 24, pp. 122–136, 2015, doi: doi:10.1016/j.inffus.2014.09.002.

[3] M. P. Sesmero, J. M. Alonso-Weber, G. Gutierrez, and A. Sanchis, "CCE: An Approach to Improve the Accuracy in Ensembles by Using Diverse Base Learners," in *International Conference on Hybrid Artificial Intelligence Systems, HAIS 2014*, 2014, pp. 630–641, doi: 10.1007/978-3-319-07617-1_55.

[4] E. L. Allwein, R. E. Schapire, and Y. Singer, "Reducing Multiclass to Binary: A Unifying Approach for Margin Classifiers," *J. Mach. Learn. Res.*, vol. 1, pp. 113–141, 2000.

[5] L. Breiman, "Bagging Predictors," *Mach. Learn.*, vol. 24, pp. 123–140, 1996, Accessed: May 04, 2012. [Online]. Available: http://www.springerlink.com/index/L4780124W2874025.pdf.

[6] T. G. Dietterich and G. Bakiri, "Solving Multiclass Learning Problems via Error-Correcting Output Codes," *J. Artif. Intell. Res.*, vol. 2, pp. 263–286, 1995, doi: doi:10.1613/jair.105.

[7] G. I. Webb, "MultiBoosting: a technique for combining boosting and wagging," *Mach. Learn.*, vol. 40, no. 2, pp. 159–196, 2000, doi: 10.1023/A:1007659514849.

[8] J. Friedman, "Another approach to polychotomous classifcation," 1996. Accessed: May 04, 2012. [Online]. Available: http://www.cba.ua.edu/~mhardin/poly.pdf.

[9] E. Alpaydin, "Combined 5 × 2 cv F Test for Comparing Supervised Classification Learning Algorithms," *Neural Comput.*, vol. 11, pp. 1885–1892, 1999, Accessed: May 04, 2012. [Online]. Available: http://www.mitpressjournals.org/doi/abs/10.1162/089976699300016007.

[10] J. Demšar, "Statistical Comparisons of Classifiers overMultiple Data Sets," *J. Mach. Learn. Res.*, vol. 7, pp. 1–30, 2006, Accessed: Feb. 05, 2013. [Online]. Available: http://dl.acm.org/citation.cfm?id=1248548.

[11] H. Chen, "Diversity and Regularization in Neural Network Ensembles," Birmingham, 2008.

[12] R. E. Banfield, L. O. Hall, K. W. Bowyer, and W. P. Kegelmeyer, "Ensemble diversity measures and their application to thinning," *Inf. Fusion*, vol. 6, no. 1, pp. 49–62, 2005, doi: 10.1016/j.inffus.2004.04.005.

[13] A. Tsymbal, M. Pechenizkiy, and P. Cunningham, "Diversity in Search Strategies for Ensemble Feature Selection," *Inf. Fusion*, vol. 6, no. 1, pp. 83–98, Mar. 2005, doi: 10.1016/j.inffus.2004.04.003.

Reliable and Fast Recurrent Neural Network Architecture Optimization

Andrés Camero
German Aerospace Center (DLR)
Munich, Germany
andres.camerounzueta@dlr.de

Jamal Toutouh
ITIS Software, Universidad de Málaga
Malaga, Spain
jamal@lcc.uma.es

Enrique Alba
ITIS Software, Universidad de Málaga
Malaga, Spain
eat@lcc.uma.es

***Abstract*—This article introduces Random Error Sampling-based Neuroevolution (RESN), a novel automatic method to optimize recurrent neural network architectures. RESN combines an evolutionary algorithm with a training-free evaluation approach. The results show that RESN achieves state-of-the-art error performance while reducing by half the computational time.**

***Index Terms*—neuroevolution, evolutionary algorithms, metaheuristics, recurrent neural networks**

I. Introduction

A recurrent neural network (RNN) is a class of artificial neural network that has feedback connections between nodes. Thanks to this recurrence, RNNs are particularly good for tackling time-dependent (or sequential) problems. But, due to this same feature (i.e., the recurrence), they are hard to train [1]: small changes on its components may produce a big performance deviation [1].

Several alternatives have been proposed to deal with the sensitivity of RNN design, including specific node architectures (e.g., GRU [2] and LSTM [3]), for their fully automatic design [4]. However, in spite of great improvements made so far [5], finding the *best* design is still an open issue. Particularly, optimizing an RNN is (time and computation wise) a very demanding task. Therefore, a *low-cost* approach is desirable.

This article summarizes our previous work [6], in which we proposed a fast and accurate method to optimize the architecture of an RNN. Particularly, we have introduced Random Error Sampling-based Neuroevolution (RESN): a $(\mu+\lambda)$ evolutionary algorithm (EA) that uses MRS [7] to guide its search, and Adam [8] to train the final solution. We have empirically validated our proposal on four prediction problems, and compared our technique to training-based architecture optimization techniques, neuroevolutionary approaches, and human expert designed solutions. The results of our experiments show that RESN achieves state-of-the-art performance, while reducing by half the overall time.

II. Random Error Sampling-based Neuroevolution

Given an input *X*, an output *Y*, a search space of RNN architectures (ARQ) and look back (or time steps, LB), we considered the problem of maximizing the *fitness(X, Y)*=p_t, the probability of finding a set of weights whose performance is below the defined threshold (i.e., MRS [7]).

To solve the stated problem, we proposed RESN, a $(\mu + \lambda)$ EA. At a glance, the *population* is a set of solutions, where each *solution* represents an RNN architecture. The initial population is randomly set by the **Initialize** function. Later, the population is assessed by the **Evaluate** function, that computes p_t for each solution. Then, the population is evolved by the selection, mutation, evaluation, replacement, and self-adjustment operations. Once the termination criterion is met, i.e., the number of *evaluations* is greater than the budget (*max_evaluations*), the **Best** solution (i.e., the one with the highest p_t) is selected and trained using Adam [8] for a predefined number of *epochs*.

Fig. 1 summarizes RESN. The proposed approach combines evolutionary computation and machine learning techniques to optimize the architecture and the weights of an RNN.

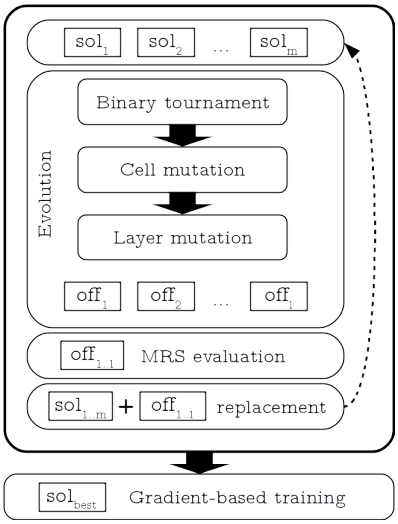


Fig. 1. The global scheme of RESN

III. Experimental Study

We tested RESN on four problems: the sine wave, the waste generation prediction problem [9], the coal-fired power plant flame intensity prediction problem [10], and the EUNITE load forecast problem [11]. Also, we proposed four experiments: **E1**: RESN vs. Gradient-based Architecture Optimization, we compared RESN against (i) a modified version of the same algorithm (i.e., a version that replaced the MRS by the results

of Adam training), (ii) the *Short training* [12] algorithm, and (iii) a *Random Search* algorithm. **E2**: RESN vs. Neuroevolution, we benchmarked RESN against EXALT [13]. **E3**: Optimization Time, we logged the execution times. And **E4**: RESN vs. Expert Design, we compared our results against the winner [11] of the EUNITE load forecast competition and to recent solutions to the same challenge [14].

Summarizing **E1**, (E1.i) the results of RESN exceed GDET (i.e., a modified version of RESN that uses training results to evaluate the performance of a solution). On average, RESN obtained a MAE of 0.105 while GDET got a 0.142 (Wilcoxon rank-sum test p-value equals to 0.001: significant). Then, (E1.ii) RESN was compared to *Short Training* [12]. The results show that there is no significant improvement in the error (i.e., Wilcoxon rank-sum test p-value equal to 0.665), however, RESN cut by half the optimization time (i.e., test **E3**). Table I presents the results, where *MAE* stands for the MAE of the final solution, and *Time* for the total time (i.e., optimization and training of the final solution) in minutes.

TABLE I
E1.II. RNN OPTIMIZATION IN THE WASTE GENERATION PREDICTION PROBLEM, A COMPARISON BETWEEN SHORT TRAINING AND RESN

	Short Training		RESN	
	MAE	Time [min]	MAE	Time [min]
Mean	0.073	97	0.079	51
Median	0.073	70	0.073	45
Max	0.076	405	0.138	103
Min	0.071	33	0.069	40
SD	0.001	75	0.017	13

To conclude E1, we compared RESN against random search (E1.iii). Despite the relatively good error performance of random search, with an average of 0.091, the Wilcoxon rank-sum test revealed that RESN (and *Short training*) beats random search (p-values are 0.017 and 0.002 respectively). Moreover, random search took nearly 50x the time of RESN (!). Thus, RESN is a fast and accurate approach (test **E3**).

Later, (E2) we compared RESN against neuroevolution. Summarizing, RESN improved EXALT by ten times (Wilcoxon rank-sum test p-value is 2.958e-06). Table II depicts the mean square error (MSE) of the solution obtained by RESN and EXALT [13] in the coal-fire power plan problem.

TABLE II
E2. RESN VS. EXALT IN THE COAL-FIRE POWER PLANT PROBLEM (MSE)

Fold	EXALT	RESN
0	0.028749	0.001541
1	0.031769	0.006536
2	0.023095	0.003821
3	0.019229	0.000570
4	0.023170	0.003336
5	0.036091	0.000617
6	0.012879	0.017061
7	0.019358	0.004032
8	0.018151	0.001912
9	0.019475	0.013996
10	0.030016	0.006120
11	0.031207	0.002942
Average	0.024432	0.005208

Finally (test **E4**), we compared RESN against human expert designed solutions in the EUNITE load forecast problem. Table III presents the results. The column SVM corresponds to [11], the winner of the "Electricity Load Forecast using Intelligent Adaptive Technologies" competition organized by EUNITE, and the other columns, i.e., BP, RBF, SVR, NNRW, KNNRW, and WKNNRW, correspond to the results presented in [14]. NA stands for *Not Available*. From the results, we concluded that the performance of RESN is comparable to a human expert, i.e., "the error of the best solution found is as good as the best solution proposed by the experts".

TABLE III
E4 RESULTS (MAPE). RESN . EXPERT DESIGN

	SVM	BP	RBF	SVR	NNRW	KNNRW	WKNNRW	RESN
Mean	2.88	NA	NA	NA	NA	NA	NA	2.28
Median	2.95	NA	NA	NA	NA	NA	NA	2.24
Max	3.48	NA	NA	NA	NA	NA	NA	3.27
Min	1.95	1.45	1.48	1.45	1.44	1.35	1.32	1.37
SD	0.01	NA	NA	NA	NA	NA	NA	0.41

IV. CONCLUSIONS AND FUTURE WORK

As a summary of this study, we conclude that RESN (which is a training-free algorithm) is a competitive approach to RNN design. Particularly, it achieves a comparable error performance of training-based RNN techniques and neuroevolutionary approaches but considerably reduces the computational time, and it is as good as human expert designed solutions.

REFERENCES

[1] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks," in *Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28*, ser. ICML 13. JMLR.org, 2013, pp. III–1310–III–1318.

[2] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translatio," *arXiv preprint arXiv:1406.1078*, 2014.

[3] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[4] J. Liang, E. Meyerson, and R. Miikkulainen, "Evolutionary architecture search for deep multitask networks," in *Proceedings of the Genetic and Evolutionary Computation Conference*, ser. GECCO '18. New York, NY, USA: ACM, 2018, pp. 466–473.

[5] V. K. Ojha, A. Abraham, and V. Snášel, "Metaheuristic design of feedforward neural networks: A review of two decades of research," *Engineering Applications of Artificial Intelligence*, vol. 60, pp. 97–116, 2017.

[6] A. Camero, J. Toutouh, and E. Alba, "Random Error Sampling-based Recurrent Neural Network Architecture Optimization," *Engineering Applications of Artificial Intelligence*, vol. 96, p. 103946, 2020.

[7] ——, "Low-Cost Recurrent Neural Network Expected Performance Evaluation," *arXiv preprint arXiv:1805.07159*, 2018.

[8] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[9] J. Ferrer and E. Alba, "Bin-ct: Urban waste collection based in predicting the container fill level," *arXiv preprint arXiv:1807.01603*, 2018.

[10] A. Ororbia, A. ElSaid, and T. Desell, "Investigating recurrent neural network memory structures using neuro-evolution," in *Proceedings of the Genetic and Evolutionary Computation Conference*, 2019, pp. 446–455.

[11] B.-J. Chen, M.-W. Chang *et al.*, "Load forecasting using support vector machines: A study on eunite competition 2001," *IEEE tran on power systems*, vol. 19, no. 4, pp. 1821–1830, 2004.

[12] A. Camero, J. Toutouh, J. Ferrer, and E. Alba, "Waste Generation Prediction in Smart Cities through Deep Neuroevolution," in *Congreso Iberoamericano de Ciudades Inteligentes (ICSC-CITIES 2018)*, 2018.

[13] A. ElSaid, S. Benson, S. Patwardhan, D. Stadem, and T. Desell, "Evolving recurrent neural networks for time series data prediction of coal plant parameters," in *International Conference on the Applications of Evolutionary Computation*. Springer, 2019, pp. 488–503.

[14] K. Lang, M. Zhang, Y. Yuan, and X. Yue, "Short-term load forecasting based on multivariate time series prediction and weighted neural network with random weights and kernels," *Cluster Computing*, pp. 1–9, 2018.

An AI-Powered System for Residential Demand Response

Iker Esnaola-Gonzalez, Francisco Javier Diez
TEKNIKER, Basque Research and Technology Alliance (BRTA)
I naki Goenaga 5, 20600 Eibar, Spain
{iker.esnaola, francisco.diez}@tekniker.es

Marko Jelić, Dea Pujić, Nikola Tomašević
Institute Mihajlo Pupin, University of Belgrade
Volgina 15, 11060 Belgrade, Serbia
{marko.jelic, dea.pujic, nikola.tomasevic}@pupin.rs

***Abstract*—Recent studies show that energy consumption of buildings has dramatically increased over the last decade, accounting for more than 35% of global energy use. However, with proper operation, significant energy savings can be achieved. Demand response is envisioned as a key enabler of this operation enhancement, as it may contribute to the reduction of demand peaks and maximization of renewable energy exploitation while mitigating potential problems with grid stability. In the article "An AI-Powered System for Residential Demand Response", a system based on artificial intelligence that solves the complex multi-objective problem to bring demand response programs to the residential sector is proposed. Through the application of novel machine learning-based algorithms, a unique control loop is developed to help dwellers determine how and when to use their appliances. The feasibility and validity of the proposed system has been demonstrated in a real-world neighbourhood where a notable reduction and shift of electricity demand peaks has been achieved. Concretely, in accordance with extreme changes in the energy prices, the users have demonstrated the ability to shift their demand to periods with lower prices as well as reducing power consumption during periods with higher prices, thus fully translating the demand peak in time. The article can be found in https://doi.org/10.3390/electronics10060693**

***Index Terms*—demand response; demand flexibility; artificial intelligence; machine learning; energy savings; peak shaving**

I. Introduction

Buildings' energy consumption has dramatically increased over the last decade due to different factors including the population growth, the increase in time spent indoors or the increased demand for building functions and indoor life quality [1]. Nevertheless, significant energy savings can be achieved in buildings if they are properly operated.

Apart from the large energy consumption of buildings, peak energy demand certainly attracts lots of attention because of its negative impacts on energy grid capital, operational costs and environmental pollution to name a few. This impact is a direct consequence of the carbon-intense generation plants that grid operators deploy in order to satisfy energy demand during these peak periods [2]. One of the resources that could contribute to significantly reduce peak demands are renewable energy sources (RES) [3], which are increasingly penetrating the energy production side. However, due to their intermittent nature, the renewable energy availability commonly does not match the distribution of energy demand in time, which may hinder their management and exploitation.

Demand-side management (DSM) activities including load curtailment (i.e., a reduction of electricity usage) and load reallocation (i.e., a shift of energy usage to other off-peak periods) have a huge potential as aids in matching energy demand with energy supply, thus avoiding these undesirable peaks. Furthermore, demand response (DR) programs are introduced into the smart grids so that reliable and economical operation of power systems are ensured [4]. DR can be understood as the set of technologies or programs that concentrate on shifting energy use short-term to help balance energy supply and demand [5]. In combination with energy generated from RES, DR is envisioned as one of the crucial enablers of curbing energy demand peaks [6].

However, the implementation of DR programs is not straightforward. The main barriers to adopt DR programs include regulatory, economic, technological and social issues [7]. Furthermore, solving the energy demand optimization for residential neighbourhoods that takes into account management of collectively shared energy assets such as RES generation as well as variable pricing tariffs and specific demand flexibility constraints is a complex multi-objective problem that requires the utilization of artificial intelligence (AI) systems. And this is where the RESPOND H2020 project (https://project-respond.eu, accessed on 2021-03-15) originates, aiming to bring DR programs to neighborhoods across Europe. With most contemporary DR implementations considering only the industrial sector, this paper proposes the simultaneous utilization of multiple different AI-based technologies such as machine learning and optimization to assist residential consumers that would like to make use of DR and incorporate it into their energy management systems. The proposed system considers the forecasted energy production and consumption based on the data acquired by the deployed Internet of Things (IoT) equipment and looks for modifications that would mitigate potential instabilities in the energy supply network by applying optimal energy utilization and load shifting. Finally, the predicted and optimal behavior are analyzed and corrective actions are suggested for dwellers, with the system even allowing for the possibility of semi-automated (remotely actuated with individual dweller consent) and fully automated (remotely actuated with previous persisting dweller consent) actions to be taken in order to minimize the additional effort that needs to be undertaken and that would disrupt daily habits.

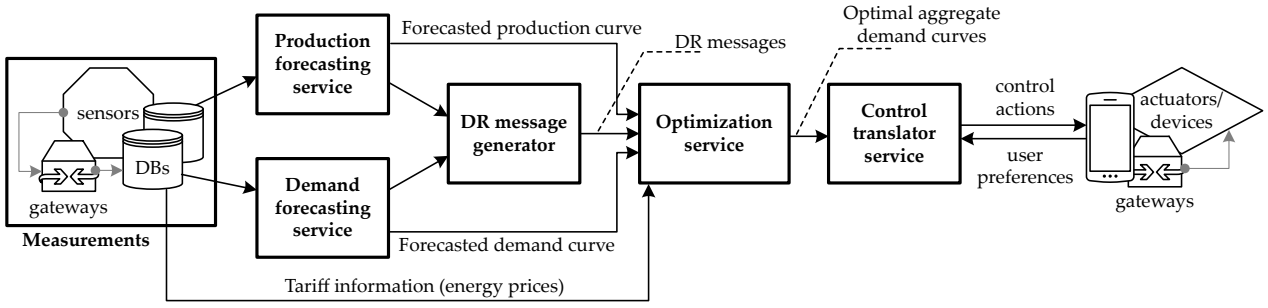


Fig. 1. The RESPOND Artificial Intelligence (AI) system loop.

This Key Work is based on the "An AI-Powered System for Residential Demand Response" article [8].

II. Contributions

RESPOND aims to allocate the most suitable demand profiles both at a dwelling and neighborhood levels as a driver for reducing the energy demand in specific time periods, as well as for maximizing the exploitation of renewable energy. To do so, RESPOND has proposed the AI system depicted in Figure 1 which has five main blocks: measurement, forecasting, demand response message generator, optimization and control block. In briefest possible terms, the measurement block is in charge of collecting and storing all the available sensor and non-sensor data. The forecasting block attempts to combine records of this data to provide projections of future production and consumption. Having in mind global issues such as demand and production peaks, a grid-responsible entity is given an opportunity to define so-called DR messages that are used as inputs to the optimization block (along the forecasted profiles) in order to guide the model towards the desired load curve. Finally, the outputs of the optimization block are analyzed by its control counterpart which suggests concrete control actions that the dwellers (end users) should carry out. With more detailed explanations of each component given in the following paragraphs, through the adequate interaction between these services, optimal dispatching of energy with regards to DR events is ensured.

In summary, the main contributions of the article are the following:

- The development of an AI-powered system for demand response in residential buildings;
- The application of the AI-powered system in a real-world neighbourhood;
- Successfully validating dwellers' willingness to adjust their loads with regards to changing energy price tariffs in a real-world neighbourhood while noting:
  - Decrease of energy demand peaks by above 30%;
  - Increase of energy demand in non-peak periods by nearly 50%;

Acknowledgment

The research presented in this paper is partly financed by the European Union (H2020 RESPOND project, Grant Agreement No.: 768619), the SPRI-Basque Government through the ELKARTEK program (3KIA project #KK-2020/00049), as well as the Ministry of Education, Science and Technological Development and the Science Fund of the Republic of Serbia (AI-ARTEMIS project, #6527051).

References

[1] X. Cao, X. Dai, and J. Liu, "Building energy-consumption status worldwide and the state-of-the-art technologies for zero-energy buildings during the past decade," *Energy and buildings*, vol. 128, pp. 198–213, 2016.
[2] L. D. Collins and R. H. Middleton, "Distributed demand peak reduction with non-cooperative players and minimal communication," *IEEE Transactions on Smart Grid*, vol. 10, no. 1, pp. 153–162, 2017.
[3] J. Jurasz and P. E. Campana, "The potential of photovoltaic systems to reduce energy costs for office buildings in time-dependent and peak-load-dependent tariffs," *Sustainable cities and society*, vol. 44, pp. 871–879, 2019.
[4] M. H. Albadi and E. F. El-Saadany, "Demand response in electricity markets: An overview," in *2007 IEEE power engineering society general meeting*. IEEE, 2007, pp. 1–5.
[5] P. Warren, "A review of demand-side management policy in the uk," *Renewable and Sustainable Energy Reviews*, vol. 29, pp. 941–951, 2014.
[6] D. Parra, S. A. Norman, G. S. Walker, and M. Gillott, "Optimum community energy storage for renewable energy and demand load management," *Applied energy*, vol. 200, pp. 358–369, 2017.
[7] N. Good, K. A. Ellis, and P. Mancarella, "Review and classification of barriers and enablers of demand response in the smart grid," vol. 72, pp. 57–72. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1364032117300436
[8] I. Esnaola-Gonzalez, M. Jelić, D. Pujić, F. J. Diez, and N. Tomašević, "An AI-Powered System for Residential Demand Response," *Electronics*, vol. 10, no. 6, p. 693, 2021.

Sobre órdenes admisibles en el conjunto de números borrosos discretos y su aplicación en problemas de toma de decisiones

Juan Vicente Riera*†, Sebastia Massanet*†, Humberto Bustince‡, Javier Fernández‡
* *Grupo de investigación en Soft Computing, Procesamiento de Imágenes y Agregación (SCOPIA)*
Departamento de Ciencias Matemáticas e Informática
Universitat de les Illes Balears, 07122 Palma, España
†*Instituto de Investigación Sanitaria de las Islas Baleares (IdISBa), 07010 Palma, España*
‡ *Departamento de Estadística, Informática y Matemáticas,*
Universidad Pública de Navarra, Campus Arrosadia, 31006 Pamplona, Spain
E-mails: jvicente.riera@uib.es, s.massanet@uib.es, bustince@unavarra.es, fcojavier.fernandez@unavarra.es

***Abstract*—Este trabajo es un resumen del artículo [1] publicado en Mathematics para su presentación en la Multiconferencia CAEPIA'21 KeyWorks.**

***Index Terms*—Número borroso discreto, Orden Total, Orden admisible, Ranking de números borrosos.**

I. Resumen

Está bien establecido en la comunidad científica que el esquema tradicional para la solución de un análisis de decisiones lingüísticas en un problema de toma de decisiones multicriterio (MCDM) consta de tres partes [2]:

1) La elección de los términos lingüísticos y/o su semántica.
2) La elección de un operador de agregación para dicha información lingüística.
3) La elección de las mejores alternativas. Este paso se divide en dos fases:
   i. Fase de agregación de la información lingüística.
   ii. Fase de explotación, en la que se establece un orden jerárquico entre las alternativas, según el valor de la interpretación lingüística colectiva (o agregada), para elegir las mejores alternativas.

De este esquema surge la necesidad de estudiar y desarrollar nuevos modelos lingüísticos computacionales que permitan representar, agregar y ordenar dicha información lingüística y que sirvan como una herramienta de apoyo para ayudar a los expertos a expresar de la manera más flexible sus preferencias [3]. Entre estos modelos lingüísticos queremos destacar, en este trabajo, el modelo lingüistico multigranular basado en números borrosos discretos [4], [5].

Como se ha mencionado anteriormente, a la hora de tomar una decisión final en un problema toma de decisiones es necesario elegir la mejor alternativa o conjunto de alternativas entre todos los alternativas disponibles y, por tanto, es necesario establecer un ordenamiento jerárquico de las mismas basadas en las preferencias dadas por los expertos.

El estudio de las relaciones de orden (totales o parciales) no es sólo un tema importante desde el punto de vista teórico, sinó que cabe destacar su relevancia en muchos problemas aplicados tales como: problemas de toma de decisión, inteligencia artificial, problemas de optimización, etc. Es obvio que la elección del orden dependerá del modelo lingüístico computacional borroso y de cada problema particular de toma de decisiones. Por esta razón, el estudio de órdenes borrosos en el contexto de los conjuntos borrosos ha sido, y sigue siendo, un tema muy candente. En particular, en el marco de los números borrosos discretos se han desarrollado pocas investigaciones y las existentes en la literatura, se basan en adaptaciones de órdenes definidos en el conjunto de números borrosos y en la utilización de índices de clasificación, los cuales, desde nuestro punto de vista, presentan algunos comportamientos no adecuados en algunas aplicaciones. Por ello, la motivación principal de nuestra investigación fue la construcción de dos familias de órdenes admisibles diferentes en el conjunto de números borrosos discretos cuyo soporte es un intervalo cerrado de una cadena finita $L_n = \{0, 1, \ldots, n\}$ que no se basan en ninguna función índice.

A continuación expondremos las definiciones y resultados más importantes para la comprensión del trabajo.

*Definición 1 (***[6]):** Sea A un conjunto no vacío, un orden parcial $\preceq$ es una relación binaria sobre A que es reflexiva, antisimétrica y transitiva. Un conjunto A con un orden parcial $\preceq$ se llama conjunto parcialmente ordenado y se denota por $(A, \preceq)$. Si en un conjunto parcialmente ordenado $(A, \preceq)$ todos sus elementos a, b son comparables, es decir, se tiene siempre que $a \preceq b$ o $b \preceq a$, el orden parcial $\preceq$ se llama orden lineal (o total) (en este caso A se llama cadena).

Denotemos por $L([0, 1])$ el conjunto de subintervalos cerrados del intervalo unidad, $L([0, 1]) = \{[a, b] \mid 0 \leq a \leq b \leq 1\}$.

*Definición 2 (***[6]):** Sea $(L([0, 1]), \preceq)$ un conjunto parcialmente ordenado. El orden $\preceq$ se llama admisible, si es un orden lineal en $L([0, 1])$ y refina el orden clásico entre intervalos, esto es, para todo $[a, b], [c, d] \in L([0, 1])$, $[a, b] \preceq [c, d]$

siempre que $[a,b] \leq_2 [c,d]$ donde $\leq_2$ denota el orden clásico entre intervalos.

Definición 3: Un subconjunto borroso A de $\mathbb{R}$ con función de pertenencia $A : \mathbb{R} \to [0,1]$ se llama *número borroso discreto*, o *dfn* de manera abreviada, si su soporte es finito, es decir, existen $x_1, ..., x_n \in \mathbb{R}$ con $x_1 < x_2 < ... < x_n$ tales que $\text{supp}(A) = \{x_1, ..., x_n\}$, y existen números naturales s, t con $1 \leq s \leq t \leq n$ tales que:

1) A(x_i)=1 para todo i con $s \leq i \leq t$. (*core*)
2) $A(x_i) \leq A(x_j)$ para todo i, j tal que $1 \leq i \leq j \leq s$.
3) $A(x_i) \geq A(x_j)$ para todo i, j tal que $t \leq i \leq j \leq n$.

Denotaremos por $\mathcal{A}_1^{L_n}$ el conjunto de los números borrosos discretos cuyo soporte es un subintevalo de la cadena finita L_n. Denotaremos por $\text{supp}(A)$ el soporte de A, es decir, el conjunto $\{x \in L_n : A(x) > 0\}$. Consideraremos $A^\alpha = \{x \in L_n : A(x) \geq \alpha\}$ para todo $\alpha \in (0,1]$ el α-corte de A. Unos α-cortes que jugarán un papel importante en los órdenes propuestos son los cortes de los llamados α-niveles relevantes.

Definición 4: Sea A un número borroso discreto tal que $\text{supp}(A) = \{x_1, ..., x_n\}$. Entonces $\alpha \in (0,1]$ es un α-nivel relevante para A si existe $x_i \in \text{supp}(A)$ tal que $A(x_i) = \alpha$.

El interés por el estudio del conjunto $\mathcal{A}_1^{L_n}$ es debido a que este tipo de subconjuntos borrosos pueden ser usados para modelar expresiones lingüísticas que describan las opiniones de los expertos en un problema de toma de decisión (ver por ejemplo, [4], [5]).

En este trabajo se proponen dos tipos de órdenes admisibles en el conjunto $\mathcal{A}_1^{L_n}$.

Definición 5: Sean $A, B \in \mathcal{A}_1^{L_n}$ dos números borrosos discretos cuyos α-niveles relevantes son $S_A = \{\alpha_1 < \cdots < \alpha_k = 1\}$ siendo $k \leq n+1$, $S_B = \{\beta_1 < \cdots < \beta_m = 1\}$ siendo $m \leq n+1$ respectivamente, y $S_{AB} = S_A \cup S_B = \{\gamma_1 < \gamma_2 < \cdots < \gamma_t = 1\}$ con $1 \leq t \leq k+m-1$. Consideremos un orden admisible $\prec_\delta$ en $\Pi[L_n] = \{[a,b] : 0 \leq a \leq b \leq n,\ a, b \in L_n\}$. Diremos que $A = B$ si, y sólo si, sus conjuntos de nivel S_{AB} son iguales, esto es, $A^{\gamma_i} = B^{\gamma_i}$ para todo $i \in I = \{1, \ldots, t\}$. Diremos que $A \prec_{\Delta_\delta^\uparrow} B$ si, y sólo si, $A \neq B$ y existe un número natural $j \in I$ tal que $A^{\gamma_j} \prec_\delta B^{\gamma_j}$ y $A^{\gamma_i} = B^{\gamma_i}$ para todo $i < j$. Diremos que $A \preceq_{\Delta_\delta^\uparrow} B$ si, y sólo si $A = B$ or $A \prec_{\Delta_\delta^\uparrow} B$.

Teorema 1: La relación binaria $\preceq_{\Delta_\delta^\uparrow}$ es un orden admisible en $\mathcal{A}_1^{L_n}$.

El segundo orden total que introdujimos sigue un patrón similar al primero, pero la comparación de los α-cortes se realiza desde la parte superior (núcleo) hasta la inferior. Definamos formalmente esta idea.

Definición 6: Sean $A, B \in \mathcal{A}_1^{L_n}$ dos números borrosos discretos cuyos α-niveles relevantes son $S_A = \{\alpha_1 < \cdots < \alpha_k = 1\}$ siendo $k \leq n+1$, $S_B = \{\beta_1 < \cdots < \beta_m = 1\}$ siendo $m \leq n+1$ respectivamente, y $S_{AB} = S_A \cup S_B = \{\gamma_1 < \gamma_2 < \cdots < \gamma_t = 1\}$ con $1 \leq t \leq k+m-1$. Consideremos un orden admisible $\prec_\delta$ en $\Pi[L_n] = \{[a,b] : 0 \leq a \leq b \leq n,\ a, b \in L_n\}$. Diremos que $A = B$ si, y sólo si, sus conjuntos de nivel en S_{AB} son iguales, esto es, $A^{\gamma_i} = B^{\gamma_i}$ para todo $i \in I = \{1, \ldots, t\}$. Diremos que $A \prec_{\Delta_\delta^\downarrow} B$ si, y sólo si, $A \neq B$ y existe un número natural $j \in I$ tal que $A^{\gamma_j} \prec_\delta B^{\gamma_j}$ y $A^{\gamma_i} = B^{\gamma_i}$ para todo $i > j$. Diremos que $A \preceq_{\Delta_\delta^\downarrow} B$ si, y sólo si, $A = B$ o $A \prec_{\Delta_\delta^\downarrow} B$.

Teorema 2: La relación binaria $\preceq_{\Delta_\delta^\downarrow}$ es un orden admisible en $\mathcal{A}_1^{L_n}$.

Finalmente queremos exponer algunas reflexiones sobre estas dos familias de ordenes admisibles en $\mathcal{A}_1^{L_n}$. Si bien ambos órdenes tienen una definición bastante similar y se basan en la comparación secuencial de los α-cortes, su interpretación desde el punto de vista de la toma de decisiones es muy diferente.

En la primera familia $\prec_{\Delta_\delta^\uparrow}$, la comparación se realiza desde el soporte hasta el núcleo y, por lo tanto, el orden prioriza α-cortes inferiores. Si los números borrosos discretos representan evaluaciones lingüísticas proporcionadas por expertos en un problema de toma de decisiones, el orden tiene en cuenta primero el soporte, que incluye todas las posibles etiquetas lingüísticas consideradas por el experto. Por tanto, este orden podría entenderse como una valoración conservadora, cualquier valor dentro del soporte se considera para tomar la decisión. Ésta es una estrategia común para generar órdenes en el conjunto de números borrosos.

Por otro lado, la segunda familia $\prec_{\Delta_\delta^\downarrow}$ decide el orden comparando los α-cortes del núcleo con el soporte priorizando α-cortes más altos. En un problema de toma de decisiones, el orden tiene en cuenta primero el núcleo, que incluye solo las etiquetas lingüísticas expresadas por el experto como aquellas con mayor valor de pertenencia. Por lo tanto, este orden podría entenderse como una evaluación más optimista, excluyendo primero cualquier etiqueta lingüística no incluida en el núcleo. En consecuencia, la elección de una de las dos familias dependerá de la actitud del grupo de expertos.

ACKNOWLEDGMENT

Este trabajo ha estado parcialmente financiado por los proyectos FEDER/Ministerio de Economía, Industria y Competitividad - AEI/TIN2016-75404-P y PID2019-108392GB-I00 (AEI/10.13039/501100011033).

REFERENCES

[1] J. V. Riera, S. Massanet, H. Bustince, J. Fernández, "On admissible orders on the set of discrete fuzzy numbers for application in Decision Making Problems," Mathematics 9(1), pp. 1–16, 2021.
[2] F. Herrera, F. Herrera-Viedma,"Linguistic decision analysis: Steps for solving decision problems under linguistic information," Fuzzy Sets Syst. 115, pp. 67—82, 2000.
[3] J. Morente-Molinera, I. Pérez, M. Ureña, E. Herrera-Viedma, " On multi-granular fuzzy linguistic modeling in group decision making problems: A systematic review and future trends," Knowl. Based Syst. 74, pp. 49–60, 2015.
[4] S. Massanet, J. V. Riera, J. Torrens, E. Herrera-Viedma, "A new linguistic computational model based on discrete fuzzy numbers for computing with words", Inf. Sci. 258, pp. 277–290, 2014.
[5] J.V. Riera, S. Massanet, E. Herrera-Viedma, J. Torrens, "Some interesting properties of the fuzzy linguistic model based on discrete fuzzy numbers to manage hesitant fuzzy linguistic information", Appl. Soft Comput.,36, pp. 383–391, 2015.
[6] L. De Miguel, H. Bustince, J. Fernandez, E. Induráin, A. Kolesárová, R. Mesiar, "Construction of admissible linear orders for interval-valued Atanassov intuitionistic fuzzy sets with an application to decision-making", Inf. Fusion 27, pp. 189–197, 2016.

Aprendizaje Federado y Privacidad Diferencial: Análisis de herramientas software, el *framework* Sherpa.ai FL y guías metodológicas para preservar la privacidad de datos

Nuria Rodríguez-Barroso
Dpto. Ciencias de Datos e IA, Instituto Interuniversitario de Investigación en Data Science and Computational Intelligence (DaSCI),
rbnuria@ugr.es

Eugenio Martínez-Cámara
Dpto. Ciencias de Datos e IA, Instituto Interuniversitario de Investigación en Data Science and Computational Intelligence (DaSCI),
emcamara@decsai.ugr.es

M. Victoria Luzón
Dpto. Lenguajes y Sismteas Inteligentes, Instituto Interuniversitario de Investigación en Data Science and Computational Intelligence (DaSCI),
luzon@ugr.es

Francisco Herrera
Dpto. Ciencias de Datos e IA, Instituto Interuniversitario de Investigación en Data Science and Computational Intelligence (DaSCI),
herrera@decsai.ugr.es

***Resumen*—Presentación del *key work* basado en el artículo *"Federated Learning and Differential Privacy: Software tools analysis, the Sherpa.ai FL framework and Methodological guidelines for preserving data privacy"* publicado en *Information Fusion*, vol. 64, p. 270-292, 2020 [1].**

***Index Terms*—Aprendizaje Federado, Privacidad Diferencial, *framework* Sherpa.ai FL**

I. Introducción

La alta demanda de servicios de inteligencia artificial en datos descentralizados que también preserven la privacidad de estos ha impulsado la investigación sobre nuevos paradigmas de aprendizaje automático que se ajusten a estos requisitos. El Aprendizaje Federado (AF) es un nuevo enfoque que protege la privacidad de los datos mediante métodos de aprendizaje que mantienen los datos en sus silos de almacenamiento.

El AF [2] se define, en su esquema más clásico, como una arquitectura cliente-servidor, en la cual los clientes $\{C_1, \ldots, C_n\}$ almacenan los datos y el servidor es un nodo de agregación. El objetivo final del AF es aprender un modelo de aprendizaje global MAG representado por los parámetros θ a partir de los modelos de aprendizaje locales entrenados en los datos de cada cliente C_i (MAL_i, representados por los parámetros θ_i) durante un proceso iterativo llamado *ronda de aprendizaje*. De esta forma, en cada ronda de aprendizaje t, cada cliente C_i entrena el MAL_i^t sobre sus datos, actualizando los parámetros de θ_i^t. Posteriormente, los parámetros globales de θ^t se calculan agregando los parámetros locales $\{\theta_1^t, \ldots, \theta_n^t\}$ con un operador de agregación Δ y se actualizan los parámetros en el MAG:

$$\begin{aligned} \theta^t &= \Delta(\theta_1^t, \ldots, \theta_n^t) \\ \theta_i^{t+1} &\leftarrow \theta^t, \quad \forall i \in \{1, \ldots, n\} \end{aligned} \tag{1}$$

Este proceso se repite de forma iterativa tantas rondas de aprendizaje como sean necesarias, de forma que θ representará el conocimiento subyacente en los clientes.

La Privacidad Diferencial (PD) [3] permite consultar un conjunto de datos sin necesidad de desvelar información sobre los datos, protegiendo de esta forma su privacidad. La combinación de la PD con el AF proporciona una protección adicional a la privacidad de los datos, que es además complementada por la capacidad de la PD de medir la pérdida de privacidad en cada ronda de aprendizaje.

El auge del AF como nuevo paradigma de aprendizaje y su combinación con la PD, como solución conjunta a los problemas de privacidad de los datos, ha impulsado la publicación de varias herramientas software para este fin. En [1] se realiza un estudio exhaustivo de estas herramientas, el cual evidencia la necesidad de un *framework* que proporcione una solución holística de AF y PD orientado a la investigación. Este *framework*, llamado Sherpa.ai Federated Learning, también se presenta en [1], y a continuación se destacan sus principales características.

II. Sherpa.ai Federated Learning: Un *framework* para AF y PD

El análsis de los *frameworks* de AF y PD evidenció un conjunto de debilidades que obligaban al desarrollo de un nuevo *framework* que combinara ambas técnicas. Esto derivó en el desarrollo de **Sherpa.ai Federated Learning (Sherpa.ai**

Tabla I: Comparación en características de AF y PD entre los software existentes y `Sherpa.ai` FL.
● Completo ● Parcial ● No funciona ● No consta

Características de AF y PD	sherpa.ai FEDERATED LEARNING	PySyft	TensorFlow	FATE	LEAF	PaddleFL
Aprendizaje Federado:						
• Utilizar modelos federados con diferentes bases de datos	●	●	●	●	●	●
• Soporte para otras librerías	●	●	●	●	●	●
• Entorno de muestreo: distribuciones IID y no-IID	●	●	●	●	●	●
• Mecanismos federados de agregación	●	●	●	●	●	●
• Simulador de ataques en AF	●	●	●	●	●	●
Privacidad Diferencial	●	●	●	●	●	●
Propiedades deseadas:						
• Documentación & tutoriales	●	●	●	●	●	●
• API de alto nivel	●	●	●	●	●	●
• Posibilidad de ampliar el marco con nuevas propiedades	●	●	●	●	●	●

FL),[1] un *framework* de código abierto que proporciona soluciones unificadas de AF y PD. Tiene como objetivo fomentar la investigación y el desarrollo de servicios de inteligencia artificial en datos descentralizados y proteger su privacidad. Sus principales características son:

1. Está diseñado siguiendo una filosofía modular, lo cual hace más intuitivo su uso, al mismo tiempo que permite la fácil incorporación de nuevas funcionalidades o la personalización de las funcionalidades existentes.
2. Diseño integrado de AF y PD.
3. Incluye la mayoría de los mecanismos de PD así como un amplio número de operadores de agregación federados. Su diseño facilita la incorporación de nuevos operadores de agregación o mecanismo de PD.
4. Soporta *scikit-learn*, *Tensorflow* y *PyTorch* para la creación de los modelos de aprendizaje.

En [1] se expone una guía metodológica sobre cómo emplear el AF para asegurar la privacidad de los datos. Esta guía, que está integrada en el diseño de Sherpa.ai FL, no habla solo de la adaptación de modelos de aprendizaje a un enfoque federado, sino de la adaptación de todo el flujo de trabajo. En primer lugar, se habla de las claves de adaptar modelos de aprendizaje distribuidos al AF dado que se incorpora la nueva restricción de la privacidad de los datos en los clientes. En segundo lugar, se habla de la adaptación de los algoritmos de aprendizaje automático que son directamente aplicables, es decir, aquellos que se pueden representar como un conjunto de parámetros que se pueden agregar según la Eq. 1. En tercer lugar, se habla de la adaptación a AF de algoritmos no directamente aplicables a este enfoque. En último lugar, se recogen una serie de recomendaciones de cómo tratar el flujo de datos de un proceso de aprendizaje automático de forma que se respeten las restricciones del AF.

Finalmente, se incluyen casos de estudio ilustrativos de cómo se emplea la funcionalidad proporcionada por Sherpa.ai FL: uno sobre clasificación en AF y otro sobre regresión haciendo uso de herramientas de AF y PD. Además, se incluyen más casos de estudio en los ejemplos del repositorio de código[2] donde se pone de manifiesto su versatilidad.

[1]https://github.com/sherpaai/Sherpa.ai-Federated-Learning-Framework

[2]https://github.com/sherpaai/Sherpa.ai-Federated-Learning-Framework/tree/master/notebooks

La tabla I resume las principales características estudiadas de los frameworks existentes en comparación con Sherpa.ai FL a fecha de Mayo de 2020. Las principales debilidades identificadas son:

1. Se centran en AF o PD, pero no proporcionan un enfoque unificado para ambos enfoques.
2. Las grandes carencias en cuanto a mecanismos de PD. Además, no permiten el desarrollo e integración de dichos mecanismos en el framework.
3. Solo incluyen los operadores de agregación más básicos.

III. Conclusiones

Como conclusión, resaltar la contribución del resultado de este trabajo, el *framework* Sherpa.ai FL, que destaca por (1) la combinación de las técnicas de AF y PD en los problemas de privacidad de datos en entornos descentralizados, (2) su enfoque de código abierto y (3) la intencionalidad de servir como herramienta de investigación proporcionando las guías metodológicas para desarrollar el flujo de trabajo de AF. Además, Sherpa.ai FL está en continuo desarrollo, por lo que se añadirán funcionalidades nuevas con el fin de que sea una herramienta útil para la investigación en AF.

Acknowledgement

Este trabajo está parcialmente subvencionado por el contrato OTRI-4137 con SHERPA Europe S.L., por el Gobierno de España bajo los proyectos TIN2017-89517-P y EQC2018-005084-P y el Fondo Europeo de Desarrollo Regional (FEDER). Nuria Rodríguez Barroso y Eugenio Martínez Cámara fueron subvencionados por los programas de ayudas del Gobierno de España Formación de Profesorado Universitario (FPU18/04475) y Juan de la Cierva Incorporación (IJC2018-036092-I) respectivamente.

Referencias

[1] N. Rodríguez-Barroso, G. Stipcich, D. Jiménez-López, J. A. Ruiz-Millán, E. Martínez-Cámara, G. González-Seco, M. V. Luzón, M. A. Veganzones, and F. Herrera, "Federated learning and differential privacy: Software tools analysis, the sherpa.ai fl framework and methodological guidelines for preserving data privacy," *Information Fusion*, vol. 64, p. 270–292, Dec 2020.

[2] Q. Yang, Y. Liu, Y. Cheng, Y. Kang, T. Chen, and H. Yu, *Federated Learning*. Morgan & Claypool, 2019, vol. 13, no. 3.

[3] C. Dwork and A. Roth, "The algorithmic foundations of differential privacy," *Foundations and Trends® in Theoretical Computer Science*, vol. 9, no. 3–4, pp. 211–407, 2014.

FEPDS: Una propuesta para la extracción de patrones emergentes difusos en flujos continuos de datos

A. M. Garcia-Vico
Instituto Andaluz Interuniversitario en Data Science and Computational Intelligence (DaSCI)
Universidad de Granada, Granada, España
agvico@decsai.ugr.es
H. Seker
Faculty of Computing, Engineering and the Build Environment, *Birmingham City University*, Birminghan, Reino Unido
huseyin.seker2@bcu.ac.uk
C. J. Carmona, P. González, M. J. del Jesus
Instituto Andaluz Interuniversitario en Data Science and Computational Intelligence (DaSCI)
Universidad de Jaén, Jaén, España
{ccarmona|pglez|mjjesus}@ujaen.es

***Resumen*—Este trabajo es un resumen del artículo publicado por los autores en la revista *IEEE Transactions on Fuzzy Systems* [1] en el que se presenta un modelo evolutivo multi-objetivo para la extracción de patrones emergentes difusos en flujos continuos de datos con gran capacidad descriptiva. Asimismo, la propuesta es aplicada en un caso de determinación en tiempo real de perfiles de los usuarios de taxis de la ciudad de Nueva York.**

***Index Terms*—Descubrimiento de reglas descriptivas supervisadas, minería de patrones emergentes, algoritmos evolutivos multi-objetivo, sistemas difusos evolutivos, minería de flujos de datos.**

I. Resumen

La minería de flujos de datos consiste en la extracción de conocimiento en una secuencia ordenada y potencialmente infinita de instancias que llegan al sistema a lo largo del tiempo a una velocidad variable [2]. Debido al incremento exponencial de dispositivos inteligentes e interconectados entre sí, este tipo de técnicas está cobrando una mayor relevancia actualmente. Sin embargo, las características de este tipo de datos suponen un reto para la extracción de conocimiento. Esto se debe principalmente a las estrictas restricciones de memoria, tiempo de cómputo y adaptación continua del modelo a la naturaleza cambiante de los datos [3].

Dentro de este ámbito, una de las aplicaciones de interés es describir o monitorizar el comportamiento de los datos respecto a una variable de interés para el experto. De este modo, se puede realizar una toma de decisiones con conocimiento del estado actual del flujo. Entre otras tareas, la minería de patrones emergentes (EPM) [4], [5] es útil para este propósito. EPM se define como la búsqueda de patrones que, dados dos conjuntos de datos, tengan un índice de crecimiento (GR) mayor a un umbral $\rho > 1$. Concretamente, consiste en describir comportamientos que se produzcan mayoritariamente en un único conjunto de datos. Esto hace que los patrones extraídos sean altamente discriminativos.

A pesar del interés que puede suscitar el conocimiento extraído por EPM, no se han encontrado métodos de este tipo para minería de flujos de datos. Aunque muchos de los métodos en la literatura emplean un esquema incremental basado en árboles [6], éstos se rigen por unos supuestos que son incompatibles con la minería de flujos de datos. Por ejemplo, asumen la disponibilidad total de los datos y no contemplan mecanismos de actualización y olvido de datos obsoletos a lo largo del tiempo. Además, la mayoría están enfocados a maximizar la precisión, ignorando por completo las bondades descriptivas de estos patrones.

En EPM es necesario que los patrones extraídos tengan, además de una gran capacidad discriminativa, gran capacidad descriptiva. Esto es especialmente relevante en minería de flujo de datos, ya que una posible aplicación es la monitorización del estado del sistema. Para ello, se debe encontrar un balance entre varios objetivos como generalidad, precisión e interés. Estos se determinan a su vez a partir de diferentes métricas, como por ejemplo WRAcc [7], entre otras. Estos objetivos son conflictivos entre sí, de modo que si aumentamos el valor de una, disminuye el valor del resto. Por tanto, las metaheurísticas multi-objetivo son adecuadas para la búsqueda de conocimiento con un buen balance entre ellos. En concreto, los métodos evolutivos han sido especialmente exitosos en la literatura para esta tarea [8], [6]. Sin embargo, el computo de estas métricas también requiere de un conjunto finito de datos para calcularas, lo que dificulta su aplicación en ámbitos de flujos continuos de datos.

Nuestra aportación a la literatura en este trabajo es un algoritmo evolutivo multi-objetivo capaz de extraer y adaptar de manera continua el conocimiento extraído en flujos de datos. El método, llamado FEPDS (*Fuzzy Emerging Patterns*

in Data Streams) tiene como principales características el procesamiento de los datos del flujo mediante bloques de tamaño fijo, un proceso de adaptación del aprendizaje basado en un enfoque ciego y un sistema evolutivo difuso multi-objetivo como método de aprendizaje con un sistema de recompensa que permite olvidar patrones antiguos. El empleo de un esquema de procesamiento por bloques de datos de tamaño fijo nos permite el cómputo directo de las métricas habituales en EPM. Por otro lado, el esquema de adaptación ciego, que ejecuta el método evolutivo por cada bloque nuevo, permite una mejor adaptación a cambios de tipo gradual [9]. Finalmente, para mejorar la adaptación, se incluye una ventana deslizante de tamaño fijo que almacena el conocimiento extraído en los m bloques de datos previos. Esta ventana será usada en el proceso evolutivo duranta la aplicación del sistema de recompensa propuesto.

El método de aprendizaje es un sistema difuso evolutivo basado en el algoritmo NSGA-II [10]. Se emplea un esquema "cromosoma=regla" en el que un individuo representa a un potencial patrón. Siguiendo la metodología de aprendizaje adaptativa, la población de individuos se inicia aleatoriamente pero incluyendo los patrones extraídos en el bloque anterior. Tras esto, el proceso evolutivo se lleva a cabo a partir del empleo de los operadores clásicos de cruce en dos puntos y mutación sesgada. Finalmente, se aplica un operador de reinicio de la población actual que también actualiza una población élite cuando el proceso se estanca. Se asume estancamiento cuando no se cubren nuevos ejemplos durante un 25 % de las iteraciones totales.

Este proceso de reinicio se basa en el procedimiento de competición de tokens [11], modificado con un esquema de recompensas. Este sistema refuerza positivamente aquellos patrones que han sido extraídos recientemente, pues la probabilidad de que el área del espacio de búsqueda cercana a estos sea prometedora es alta. Sin embargo, conforme el tiempo avanza, esta probabilidad decae debido a la naturaleza cambiante de los datos. Por tanto, se debe dar menos recompensa a los patrones extraídos con mayor antigüedad. Esto se consigue con la fórmula presentada en la Ecuación 1.

$$Div_t(fEP_i) = WRAcc_t(P_i) + \sum_{j=t-m}^{t} SW(P_i, j) \cdot 2^{-(t-j)} \cdot WRAcc_j(P_i) \quad (1)$$

donde $WRAcc_t$ es el valor de WRAcc en el instante t y SW es una función que devuelve uno si el patrón P_i fue extraído en el instante j o cero en caso contrario. Nótese que para poder usar SW es necesario determinar una ventana deslizante que almacene los patrones extraídos en los m bloques de datos anteriores.

La validez del método propuesto se analiza a través de un exhaustivo estudio experimental formado por 94 flujos de datos artificiales con diferentes características de cambio, en donde se analizan la calidad de los patrones extraídos, su capacidad de adaptación al cambio de concepto y su tiempo de cómputo y escalabilidad. Los resultados demuestran que el método propuesto es capaz de procesar bloques de hasta 15000 instancias en dos segundos. La calidad media de los patrones extraídos es buena para realizar un análisis rápido y de calidad. Finalmente, se demuestra que el método presenta una capacidad rápida de adaptación del conocimiento a los cambios en los datos.

Tras determinar las bondades del método sobre datos artificiales, se aplica el algoritmo propuesto en un caso real. El objetivo de este estudio es doble: por un lado se pretende obtener los perfiles de usuario en tiempo real de los taxis de la ciudad de Nueva York. Por otro lado, con el conocimiento extraído y almacenado a lo largo del tiempo, se pretenden extraer perfiles de usuario más generales. Los resultados de este estudio muestran una gran calidad en la extracción de conocimiento en tiempo real aplicable a niveles operativos, mientras que también puede usarse para realizar análisis de calidad a largo plazo en niveles niveles estratégicos.

AGRADECIMIENTOS

Este trabajo ha sido subvencionado por la Consejería de Transformación Económica, Industria, Conocimiento y Universidades de la Junta de Andalucía, programa Personal Investigador Doctor, referencia DOC_00235.

REFERENCIAS

[1] A. García-Vico, C. J. Carmona, P. González, and M. J. del Jesus, "Fepds: A proposal for the extraction of fuzzy emerging patterns in data streams," *IEEE Transactions on Fuzzy Systems*, vol. 28, no. 12, pp. 3193–3203, 2020.

[2] J. Gama, *Knowledge discovery from data streams*. CRC Press, 2010.

[3] I. Khamassi and M. Sayed Mouchaweh, "Drift detection and monitoring in non-stationary environments," in *2014 IEEE Conference on Evolving and Adaptive Intelligent Systems, EAIS 2014, Linz, Austria, June 2-4, 2014*, 2014, pp. 1–6.

[4] G. Dong and J. Li, "Efficient mining of emerging patterns: Discovering trends and differences," in *Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, NY, USA: ACM, 1999, pp. 43–52.

[5] A. M. García-Vico, C. J. Carmona, D. Martín, M. García-Borroto, and M. J. del Jesus, "An overview of emerging pattern mining in supervised descriptive rule discovery: Taxonomy, empirical study, trends and prospects," *WIREs: Data Mining and Knowledge Discovery*, vol. 8, no. 1, 2018.

[6] A. M. García-Vico, C. J. Carmona, P. González, and M. J. del Jesus, "MOEA-EFEP: Multi-objective evolutionary algorithm for extracting fuzzy emerging patterns," *IEEE Transactions on Fuzzy Systems*, vol. 26, no. 5, pp. 2861 – 2872, 2018.

[7] C. J. Carmona, M. J. del Jesus, and F. Herrera, "A Unifying Analysis for the Supervised Descriptive Rule Discovery via the Weighted Relative Accuracy," *Knowledge-Based Systems*, vol. 139, pp. 89–100, 2018.

[8] F. Pulgar-Rubio, A. J. Rivera-Rivas, M. D. Pérez-Godoy, P. González, C. J. Carmona, and M. J. del Jesus, "MEFASD-BD: Multi-Objective Evolutionary Fuzzy Algorithm for Subgroup Discovery in Big Data Environments - A MapReduce Solution," *Knowledge-Based Systems*, vol. 117, pp. 70–78, 2017.

[9] I. Khamassi, M. Sayed Mouchaweh, M. Hammami, and K. Ghédira, "Discussion and review on evolving data streams and concept drift adapting," *Evolving Systems*, vol. 9, no. 1, pp. 1–23, 2018.

[10] K. Deb, A. Pratap, S. Agrawal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.

[11] K. S. Leung, Y. Leung, L. So, and K. F. Yam, "Rule Learning in Expert Systems Using Genetic Algorithm: 1, Concepts," in *Proc. of the 2nd International Conference on Fuzzy Logic and Neural Networks*, K. Jizuka, Ed., 1992, pp. 201–204.

Miniaturización de aplicaciones JavaScript para el Internet de las Cosas

Rubén Saborido*, Rodrigo Morales†, Yann-Gaël Guéhéneuc†
*ITIS Software, Universidad de Málaga, España
†Computer Science and Software Engineering, Concordia University, Canadá
Email: rsain@uma.es, rodrigomorales2@acm.org, yann-gael.gueheneuc@concordia.ca

***Resumen*—El Internet de las Cosas (IdC) se refiere a la interconexión de dispositivos digitales cotidianos con Internet. Estos dispositivos se conectan a distintos servidores Web y otros dispositivos con los que periódicamente intercambian información a través de la red. Uno de los lenguajes de programación para la comunicación entre servicios software en la nube y el desarrollo de aplicaciones Web es JavaScript (JS). Sin embargo, muchos dispositivos en el IdC, como los *Systems-on-a-Chip* (SoC), poseen un hardware restrictivo: reducida capacidad de almacenamiento, escasa memoria RAM y CPU poco potente. Debido a estas limitaciones, puede no ser trivial portar una aplicación JS existente a diferentes dispositivos de este tipo. Según lo anterior, las empresas suelen migrar estas aplicaciones a otros lenguajes para poder ejecutar este software en distintas plataformas hardware. Sin embargo, esta solución es costosa e implica mantener diferentes implementaciones de una misma aplicación.**

MoMIT es una herramienta software que mediante técnicas de Inteligencia Artificial (IA) permite miniaturizar aplicaciones JS para ser ejecutadas en dispositivos del IdC. Este proceso de miniaturización no modifica el código fuente, consiste en deshabilitar aquellas características o *flags* del intérprete JS que no son necesarias para ejecutar una aplicación determinada. Esta herramienta se ha ejecutado sobre un popular intérprete de JS y se ha evaluado sobre 23 aplicaciones, antes y después de su miniaturización. En base a los experimentos realizados, MoMIT es capaz de reducir en un 31 %, 56 % y 36 %, el tamaño, el uso de memoria y el de CPU, respectivamente, de aplicaciones JS.

***Index Terms*—Optimización, Algoritmos Evolutivos, Internet de las Cosas, Miniaturización de Software, JavaScript.**

I. Introducción

El Internet de las Cosas (IdC) es una red de dispositivos interconectados que proveen un determinado servicio a través de Internet. Uno de los lenguajes de programación más extendido en la Web es JavaScript (JS), el cual lleva ocho años siendo el lenguaje de programación más usado.[1] Esto es en gran parte debido a que muchas empresas tecnológicas han desarrollado sus productos software en JS en los últimos años. Este software se ha diseñado generalmente para su ejecución en navegadores Web de usuarios con hardware ordinario (ordenadores de escritorio, portátiles, teléfonos móviles y tablets). Sin embargo, debido a la rapida expansión del IdC, muchas de estas empresas desean en la actualidad portar estas aplicaciones a dispositivos *Systems-on-a-Chip* (SoC), como Arduino Uno[2]. Para ello, disponen de dos posibles alternativas. Una opción es migrar el código de la aplicación JS a un lenguaje de más bajo nivel, como C. Pero esto supone que el equipo de desarrollo debe mantener dos versiones del código, una en JS y otra en C. Una segunda alternativa es empaquetar un intérprete de JS junto a la aplicación original. Sin embargo, esta última opción puede no ser viable debido principalmente a las limitaciones hardware de estos dispositivos (CPU, memoria y capacidad de almacenamiento masivo). Aquí es donde entra MoMIT: una herramienta que permite portar aplicaciones JS a dispositivos SoC. Esto se hace integrando un intérprete de JS que es personalizado mediante técnicas de Inteligencia Artificial (IA) para optimizar el rendimiento de aplicaciones JS.

El presente *key work* corresponde al artículo "MoMIT: Porting a JavaScript Interpreter on a Quarter Coin" [1], publicado en 2020 en la revista IEEE Transactions on Software Engineering.

II. Objetivo

Este trabajo propone una herramienta software que permite miniaturizar aplicaciones Web desarrolladas en JS para ser ejecutadas en dispositivos SoC. Esto se hace empaquetando un intérprete de JS, como Duktape[3], junto a la aplicación original. Duktape permite configurar diferentes *flags* o características al empaquetar una aplicación JS. Estos *flags* se usan, por ejemplo, para deshabilitar todo lo relacionado con expresiones regulares, el soporte para procesamiento de textos en formato JSON, etc. Desactivando aquellas características del intérprete que no sean requeridas para la correcta ejecución de una aplicación puede mejorarse de forma significativa el rendimiento de ésta, permitiendo incluso su despliegue y ejecución en dispositivos SoC. A este proceso es al que se denomina miniaturización.

III. Problema

Formulamos el problema de miniaturizar una aplicación JS como un problema de optimización multiobjetivo. Se parte de un intérprete JS y de una lista de especificaciones hardware de diferentes dispositivos donde se desea ejecutar la aplicación. El problema consiste en encontrar aquella combinación de *flags* del intérprete JS que permita ejecutar correctamente la aplicación proporcionada ajustándose a las limitaciones hardware de distintos dispositivos mientras minimiza diferentes funciones objetivo. En nuestro estudio consideramos

[1]https://insights.stackoverflow.com/survey/2020
[2]https://store.arduino.cc/arduino-uno-rev3
[3]https://duktape.org

las siguientes funciones objetivo a minimizar: uso de CPU, memoria RAM consumida y espacio en disco usado. No obstante, podrían incluirse otras funciones objetivo, como el consumo de energía y/o el uso de datos.

Cada aplicación JS requiere ciertos *flags* activos en el intérprete para poder ejecutarse correctamente. A este conjunto de *flags* lo denominamos $ComF$. El resto de *flags* son opcionales y cada uno de ellos tiene un impacto concreto en las funciones objetivo anteriores. A este conjunto de *flags* lo denominamos OF. El número de configuraciones posibles es 2^{OF}. Lo cual puede hacer el problema de minituarización de aplicaciones JS inmanejable, especialmente en su resolución de forma exacta.

IV. Solución

MoMIT es una herramienta software que toma como entrada (1) un intérprete JS y una aplicación JS, (2) el conjunto de *flags* F configurables en el intérprete dado, (3) una lista L de aquellos *flags* que deben estar activados para el correcto funcionamiento de la aplicación proporcionada ($L \subseteq F$) y (4) una lista de las especificaciones hardware de aquellos dispositivos a donde desea portarse la aplicación. MoMIT hace uso de técnicas de Inteligencia Artificial para encontrar combinaciones de *flags* que satisfacen las restricciones de los dispositivos proporcionados para las distintas funciones objetivo.

V. Resultados

MoMIT permite miniaturizar aplicaciones JS para ser ejecutadas en distintos dispositivos SoC sin necesidad de modificar el código fuente. Esto es de gran relevancia para aquellas empresas que desean portar su software desarrollado en JS para ser desplegado en el IdC.

MoMIT se ha ejecutado sobre el intérprete de JS Duktape y se ha evaluado sobre 23 aplicaciones, antes y después de la miniaturización. Los resultados son muy prometedores, reduciendo el tamaño de estos programas, su uso de memoria y de CPU, en un 31 %, 56 % y 36 %, respectivamente. De estas 23 aplicaciones, 21 (91 %) han podido ejecutarse satisfactoriamente en dispositivos con hardware muy limitado, como ESP32[4] que posee un tamaño similar al de una moneda.

Esta técnica puede ser aplicable a otros lenguajes de programación interpretados, como Python. Como trabajo futuro queremos extender MoMIT para soportar este lenguaje de programación haciendo uso de CPython como intérprete. Esto haría la herramienta mucho más versatil, cubriendo dos de los lenguajes de programación más usados en la actualidad.

Agradecimientos

El trabajo original fue soportado por el Natural Sciences and Engineering Research Council of Canada (NSERC). Actualmente, Rubén Saborido es beneficiario de la beca Juan de la Cierva - Formación (referencia FJC2018-038537-I).

[4]http://esp32.net

Referencias

[1] R. Morales, R. Saborido, and Y.-G. Guéhéneuc, "Momit: Porting a javascript interpreter on a quarter coin," *IEEE Transactions on Software Engineering*, pp. 1–1, 2020.

E2PAMEA: un algoritmo evolutivo para la extracción eficiente de patrones emergentes difusos en entornos big data

A.M. Garcia-Vico
Instituto Andaluz Interuniversitario en Data Science and Computational Intelligence (DaSCI)
Universidad de Granada, Granada, España
agvico@decsai.ugr.es
D. Elizondo
School of Computer Science and Informatics
DeMonfort University, Leicester, Reino Unido
elizondo@dmu.ac.uk
F. Charte, P. González, C. J. Carmona
Instituto Andaluz Interuniversitario en Data Science and Computational Intelligence (DaSCI)
Universidad de Jaén, Jaén, España
{fcharte|pglez|ccarmona}@ujaen.es

***Resumen*—Este trabajo es un resumen del artículo publicado por los autores en la revista *Neurocomputing* [1] en el que se presenta un modelo evolutivo multi-objetivo adaptativo junto a un sistema de evaluación eficiente para la extracción de patrones emergentes difusos en entornos de datos masivos.**

***Index Terms*—Descubrimiento de reglas descriptivas supervisadas, minería de patrones emergentes, algoritmos evolutivos multi-objetivo, sistemas difusos evolutivos, Big Data.**

I. Resumen

La cantidad de información que se genera actualmente crece de manera exponencial y habitualmente no puede ser procesada por una única máquina, por lo que la aplicación de técnicas de computación distribuida es obligatoria. A esta cantidad ingente de datos se denomina en la literatura como *big data* [2].

Dentro de este ámbito, la extracción de conocimiento capaz de describir, de manera comprensible, el comportamiento de los datos respecto a una variable de interés para el experto es fundamental. Entre otras tareas, la minería de patrones emergentes (EPM) [3], [4] es útil para este propósito. EPM se define como la búsqueda de patrones que, dados dos conjuntos de datos D_1 y D_2, tengan un índice de crecimiento (GR) mayor a un umbral $\rho > 1$. Concretamente, consiste en describir comportamientos que se produzcan mayoritariamente en un único conjunto de datos. Esto hace que los patrones extraídos sean muy discriminativos. Para calcular el GR, es necesario determinar qué instancias cumplen las características descritas por un patrón en concreto, para posteriormente poder determinar su calidad. Esto se lleva a cabo a través del cálculo de una matriz de contingencia de cuatro valores (tp, fp, fn, tn) que determinan el número de instancias cubiertas/no cubiertas de manera correcta/incorrecta, respectivamente. Con estos cuatro valores, el GR se define como $GR = \frac{tp(fp+tn)}{fp(tp+fn)}$.

No obstante, en EPM es necesario que los patrones extraídos tengan, además de una gran capacidad discriminativa, gran capacidad descriptiva. Para ello, se debe encontrar un balance entre varios objetivos como generalidad, precisión e interés. Estos se determinan a su vez a partir de diferentes métricas, como por ejemplo WRAcc [5], entre otras. Estos objetivos son conflictivos entre sí, de modo que si aumentamos el valor de una, disminuye el valor del resto. Por tanto, las metaheurísticas multi-objetivo son adecuadas para la búsqueda de conocimiento con un buen balance entre ellos. En concreto, los métodos evolutivos han sido especialmente exitosos en la literatura para esta tarea [6], [7].

A pesar de sus buenos resultados, el principal problema de estas técnicas es su escalabilidad. Esto se debe principalmente a que para calcular estas métricas objetivo, se requiere determinar para cada patrón candidato su matriz de contingencia asociada. Para obtenerla se necesita recorrer todo el conjunto de datos completo, lo que en entornos *big data* es muy costoso. El paradigma MapReduce [8] es la estrategia más habitual en la literatura para aliviar este problema. Dentro de este paradigma, se han empleado dos enfoques diferentes: 1) un enfoque local, donde se ejecuta un método no distribuido en cada partición de datos y 2) un enfoque global, donde una parte, o el método completo, se rediseña para su trabajo de manera distribuida. Los enfoques locales son fácilmente escalables, pero el diseño del método de agregación no es trivial y su calidad depende de cómo se han distribuido los datos en cada partición. Por otro lado, los enfoques globales habitualmente permiten extraer el resultado exacto independientemente del particionado empleado. Normalmente, un enfoque global es deseable respecto a un enfoque local. Sin embargo, su diseño suele ser más complejo y su tiempo de ejecución más lento.

Nuestra aportación a la literatura en este trabajo es doble:

por un lado se presenta un método de evaluación distribuido denominado Bit-LUT junto a un método evolutivo multi-objetivo de extracción de patrones emergentes difusos denominado E2PAMEA. El método Bit-LUT se fundamenta en la premisa de que las definiciones de las etiquetas lingüísticas difusas no se modifican durante el proceso evolutivo, por lo que se puede precalcular el grado de pertenencia de cada ejemplo a cada etiqueta para mejorar la eficiencia. En concreto, Bit-LUT almacena para cada par variable-valor (i, j) del problema un vector binario BS_i^j que indica si una instancia k es cubierta o no por dicho par. Este proceso de precálculo se lleva a cabo mediante un proceso MapReduce, en donde cada partición calcula un subconjunto de este vector BS_i^j en la fase *map*, mientras que la fase *reduce* únicamente concatena estos vectores para obtener el resultado final.

Por otro lado, E2PAMEA es un enfoque evolutivo multi-objetivo basado en un enfoque "cromosoma = regla" en el que un individuo de la población representa un potencial patrón. El resultado final es el conjunto formado por la unión de varios individuos. Estos interactúan entre sí mediante el empleo de un enfoque cooperativo-competitivo en donde los individuos compiten entre sí debido al empleo de los operadores genéticos mientras que, por otro lado, colaboran en la búsqueda del mejor conjunto de patrones final gracias al empleo de una población élite. En esta población élite se aplica el operador de competición de tokens [9] para buscar la población élite con el valor de atipicidad medio más elevado. Respecto a los operadores genéticos empleados, se emplean dos operadores de cruce: cruce en dos puntos y HUX; y dos operadores de mutación: cambio aleatorio de un gen y eliminación al azar de una variable completa. Estos operadores se aplican con una probabilidad adaptativa en función del número de individuos generados que han sobrevivido de una generación a la siguiente. De este modo, se evita la necesidad de optimizar manualmente estos parámetros. Para llevar a cabo una evaluación eficiente E2PAMEA hace uso de Bit-LUT para determinar los ejemplos cubiertos por un individuo a través del proceso mostrado en la Ecuación (1).

$$Cov(P_i) = \left(BS_1^1 \mid BS_1^2 \mid \ldots \mid BS_1^j\right) \& \ldots \& \left(BS_i^1 \mid BS_i^2 \mid \ldots \mid BS_i^j\right) \quad (1)$$

donde $\mid$ y $\&$ son las operaciones de bits OR y AND, respectivamente. Esta operación se distribuye a su vez mediante un proceso MapReduce. Para ello, la fase *map* se encarga del cómputo de las operaciones OR para cada variable del problema, mientras que la fase *reduce* se encargaría de aplicar todas las operaciones AND para obtener el resultado final. Para mejorar la eficiencia este cómputo se realiza para todos los individuos de la población.

La validez del método propuesto se estudia a través de un exhaustivo estudio experimental en el que se emplean 6 conjuntos de datos de gran dimensionalidad en el que se busca determinar la calidad del conocimiento extraído y el tiempo de ejecución empleado con respecto al método BD-EFEP [10], uno de lo métodos evolutivos más prometedores del ámbito. Finalmente, se determina la escalabilidad de la propuesta respecto a ambas dimensiones del conjunto de datos.

En el estudio experimental realizado se demuestra que la calidad del conocimiento extraído es más precisa, más interesante y más general en E2PAMEA que en BD-EFEP, debido principalmente al enfoque adaptativo propuesto. Por otro lado, el tiempo de ejecución se reduce en hasta 60 veces respecto a BD-EFEP, mientras que su consumo de memoria se reduce en hasta 10 veces gracias principalmente al empleo de conjuntos de bits y operaciones bits, que son altamente eficientes. Finalmente, como resultado del análisis de escalabilidad empleado se determina que el coste computacional del método propuesto aumenta de manera lineal respecto al número de instancias y variables. En concreto, se han conseguido procesar hasta 10^8 instancias con 100 pares variable-valor y 10^6 instancias con hasta 75000 pares variable-valor en aproximadamente 5000 segundos.

AGRADECIMIENTOS

Este trabajo ha sido subvencionado por la Consejería de Transformación Económica, Industria, Conocimiento y Universidades de la Junta de Andalucía, programa Personal Investigador Doctor, referencia DOC_00235.

REFERENCIAS

[1] A. García-Vico, F. Charte, P. González, D. Elizondo, and C. J. Carmona, "E2pamea: A fast evolutionary algorithm for extracting fuzzy emerging patterns in big data environments," *Neurocomputing*, vol. 415, pp. 60 – 73, 2020.

[2] M. A. Beyer and D. Laney, "The importance of big data: a definition," *Gartner Research Report*, pp. 1–9, 2012.

[3] G. Dong and J. Li, "Efficient mining of emerging patterns: Discovering trends and differences," in *Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, NY, USA: ACM, 1999, pp. 43–52.

[4] A. M. García-Vico, C. J. Carmona, D. Martín, M. García-Borroto, and M. J. del Jesus, "An overview of emerging pattern mining in supervised descriptive rule discovery: Taxonomy, empirical study, trends and prospects," *WIREs: Data Mining and Knowledge Discovery*, vol. 8, no. 1, 2018.

[5] C. J. Carmona, M. J. del Jesus, and F. Herrera, "A Unifying Analysis for the Supervised Descriptive Rule Discovery via the Weighted Relative Accuracy," *Knowledge-Based Systems*, vol. 139, pp. 89–100, 2018.

[6] F. Pulgar-Rubio, A. J. Rivera-Rivas, M. D. Pérez-Godoy, P. González, C. J. Carmona, and M. J. del Jesus, "MEFASD-BD: Multi-Objective Evolutionary Fuzzy Algorithm for Subgroup Discovery in Big Data Environments - A MapReduce Solution," *Knowledge-Based Systems*, vol. 117, pp. 70–78, 2017.

[7] A. M. García-Vico, C. J. Carmona, P. González, and M. J. del Jesus, "MOEA-EFEP: Multi-objective evolutionary algorithm for extracting fuzzy emerging patterns," *IEEE Transactions on Fuzzy Systems*, vol. 26, no. 5, pp. 2861 – 2872, 2018.

[8] N. Desai and A. Ganatra, "Incorporating boundary value concept and recency constraint to capture emerging trends in time stamp based sequence dataset," in *Proc. of the 2015 International Conference on Communication, Information and Computing Technology*, 2015, pp. 1–7.

[9] K. S. Leung, Y. Leung, L. So, and K. F. Yam, "Rule Learning in Expert Systems Using Genetic Algorithm: 1, Concepts," in *Proc. of the 2nd International Conference on Fuzzy Logic and Neural Networks*, K. Jizuka, Ed., 1992, pp. 201–204.

[10] A. M. García-Vico, C. J. Carmona, P. González, and M. J. del Jesus, "A big data approach for extracting fuzzy emerging patterns," *Cognitive Computation*, vol. 11, no. 3, pp. 400 – 417, 2019.

Fostering Diversity in Spatial Evolutionary Generative Adversarial Networks

Jamal Toutouh
Massachusetts Institute of Technology
Cambridge, MA, USA
toutouh@mit.edu

Erik Hemberg
Massachusetts Institute of Technology
Cambridge, MA, USA
hembergerik@csail.mit.edu

Una-May O'Reilly
Massachusetts Institute of Technology
Cambridge, MA, USA
unamay@csail.mit.edu

***Abstract*—Generative adversary networks (GANs) suffer from training pathologies such as instability and mode collapse, which mainly arise from a lack of diversity in their adversarial interactions. Co-evolutionary GAN (CoE-GAN) training algorithms have shown to be resilient to these pathologies. This article introduces Mustangs, a spatially distributed CoE-GAN, which fosters diversity by using different loss functions during the training. Experimental analysis on MNIST and CelebA demonstrated that Mustangs trains statistically more accurate generators.**

***Index Terms*—generative adversarial networks, co-evolutionary algorithms, neural networks**

I. Introduction

Generative adversarial networks (GANs) have emerged as a powerful machine learning paradigm for the task of estimating a distribution function underlying a given set of samples [1]. The successes of GANs in generating realistic, complex, multivariate distributions motivated a growing body of applications, such as realistic image generation [2], video prediction [3], and text to image synthesis [4].

A GAN consists of two neural networks (NN), a generator and a discriminator, which apply adversarial learning to train their weights against each other (formulated as a minmax optimization problem). The discriminator is trained to correctly discern the "natural/real" samples from "artificial/fake" samples produced by the generator. The generator is trained to transform a random input into samples that fool the discriminator. GANs are notoriously hard to train, frequently showing pathologies such as mode collapse or vanishing gradients [5], [6].

Co-evolutionary training (CoE) has shown to be resilient to the GAN training degenerative behaviors by evolving two populations of NNs (one of generators against one of discriminators) towards convergence while keeping genome diversity [6]. This article summarizes our previous research [7], in which we proposed a CoE GAN training method named MUtation SpaTial gANs (**Mustangs**). **Mustangs** combines the ideas of two successful GAN training approaches that foster diversity by applying different strategies: Evolutionary GAN (E-GAN), which injects diversity by applying three different NN mutations [5], and Lipizzaner, which applies a spatially distributed cellular CoE traiining [8]. The main aim of this research was to show that combining ideas from both training methods improve diversity, and therefore, is better than either one of them.

II. MUtation SpaTial GANs training

Mustangs applies a spatially distributed CoE to train GANs. It evolves two populations, $P_u = \{u_1, \ldots, u_T\}$ a population of generators and $P_v = \{v_1, \ldots, v_T\}$ a population of discriminators to create diversity in genomes spaces. Both populations (P_u and P_v) are distributed on the cells of a two dimensional toroidal grid [6]. The fitness $\mathcal{L}$ of each generator $u_i \in P_u$ and discriminator $v_j \in P_v$ are assessed according to their interactions with a set of discriminators from P_v and generators from P_u, respectively. The fittest individuals are used to generate the new of individuals (generators and discriminators) by applying mutation. The new individuals replace the ones in the current population if they perform better (better fitness) to produce the next generation.

Instead of applying the CoE in an *all-vs-all* flavor, the cell's *neighborhood* defines the subset of individuals (sub-populations) of P_u and P_v to interact with and it is specified by its size s_n. In our study, we use a five-cell neighborhood, i.e, one center and four adjacent cells. In **Mustangs**, each neighborhood (sub-population) performs an instance of the CoE to update its center cell with the fittest NN after each training epoch. Besides, the fittest individuals are sent to the neighborhood cells to update the sub-populations of the grid.

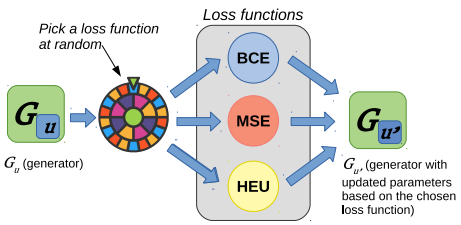


Fig. 1. Mutation used in **Mustangs**.

At the time of writing the article, CoE applied the mutation based on stochastic gradient descent (SGD) to create the offspring by minimizing only one objective/loss function, which generally attempts to minimize the distance between the generated fake data and real data distributions according to a given measure [8]. Instead, **Mustangs** generates the offspring by an applying SGD-based mutation that randomly picks a giving training objective function (see Fig.1). **Mustangs** applies mutations introduced by E-GAN [5]: *Minmax mutation* (BCE), *Least-square mutation* (LSE), and *Heuristc mutation* (HEU).

III. Experimental analysis

This section summarizes the main outcomes of the empirical analysis shown in our previous work [7]. **Mustangs** was evaluated on two common image data sets: MNIST and CelebA. The experiments took into account: **GAN-BCE**, a standard GAN which uses BCE objective; E-GAN; a spatial CoE that applies each one of the objective functions, **Lip-BCE**, **Lip-MSE**, and **Lip-HEU**; and **Mustangs**. All evaluated methods used the same computational budget.

The Frechet inception distance (FID) was evaluated to asses the accuracy of the generated fake data [9] and the total variation distance (TVD) for diversity.

Table I shows the best FID values on MNIST. **Mustangs** has the lowest (best) median. All the spatially distributed methods are better than **E-GAN** and **GAN-BCE**. The results indicate that **Mustangs** is robust to the varying performance of the individual loss functions (lowest deviation). This helps to strengthen the idea that diversity, both in genome and mutation space, provides robust GAN training. A ranksum test with Holm correction confirms that the difference between **Mustangs** and the other methods is significant at confidence levels of α <0.01.

TABLE I
FID MNIST results(Low FID indicates good performance)

Algorithm	Mean	Std%	Median	IQR
Mustangs	*42.235*	12.863%	*43.181*	7.586
Lip-BCE	48.958	20.080%	46.068	4.663
Lip-MSE	371.603	20.108%	381.768	104.625
Lip-HEU	52.525	17.230%	52.732	9.767
E-GAN	466.111	10.312%	481.610	69.329
GAN-BCE	457.723	2.648%	459.629	17.865

Table II summarizes TVD results on MNIST. The methods that provide genome diversity generate more diverse data samples than the other two analyzed methods. The three algorithms with the lowest (best) FID score (**Mustangs**, **Lip-BCE**, and **Lip-HEU**) also provide the lowest (best) TVD values. The best TVD result is obtained by **Lip-HEU**.

TABLE II
MNIST TVD results (Low TVD indicates more diversity)

Alg.	Mustangs	Lip-BCE	Lip-HEU	Lip-MSE	E-GAN	GAN-BCE
TVD	0.180	0.171	*0.115*	0.365	0.534	0.516

Figure 2 illustrates how spatially distributed CoE algorithms produce robust generators that provide with accurate MNIST samples across all the classes.



a) **Mustangs** b) **Lip-BCE** c) **E-GAN** d) **GAN-BCE**

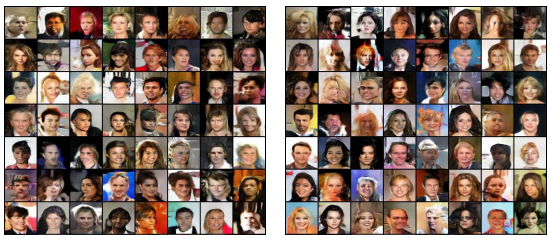
Fig. 2. Sequence of samples generated of MNIST dataset.

Table III summarizes FID the results on CelebA. **Mustangs** provides the lowest median FID. **Lip-BCE** and **Lip-HEU** provide median and mean FIDs close to the **Mustangs** ones. However, **Mustangs** is the most robust (see deviations).

Figure 3 illustrates a sequence of samples generated by the best generators in terms of FID score of the most competitive two training methods, i.e., **Mustangs** and **Lip-BCE**. As it can be seen in these two sets images generated, both methods present similar capacity of generating human faces.

TABLE III
FID CelebA results (Lower is better)

Algorithm	Mean	Std%	Median	IQR
Mustangs	36.148	0.571%	36.087	0.237
Lip-BCE	36.250	5.676%	36.385	2.858
Lip-MSE	158.740	40.226%	160.642	47.216
Lip-HEU	37.872	5.751%	37.681	2.455



(a) **Mustangs** (b) **Lip-BCE**

Fig. 3. Sequence of samples generated of CelebA.

IV. Conclusions and Future Work

The research carried out showed that GAN training can be improved by boosting diversity. Mustangs tested on the MNIST and CelebA datasets showed the best accuracy, robustness, and high diversity in label space. This research was the root of other published studies [10], [11].

References

[1] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[2] Z. Gan, L. Chen, W. Wang, Y. Pu, Y. Zhang, H. Liu, C. Li, and L. Carin, "Triangle generative adversarial networks," in *Advances in Neural Information Processing Systems*, 2017, pp. 5247–5256.

[3] X. Liang, L. Lee, W. Dai, and E. P. Xing, "Dual motion gan for future-flow embedded video prediction," in *IEEE International Conference on Computer Vision (ICCV)*, vol. 1, 2017.

[4] S. E. Reed, Z. Akata, S. Mohan, S. Tenka, B. Schiele, and H. Lee, "Learning what and where to draw," in *Advances in Neural Information Processing Systems*, 2016, pp. 217–225.

[5] C. Wang, C. Xu, X. Yao, and D. Tao, "Evolutionary generative adversarial networks," *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 6, pp. 921–934, 2019.

[6] A. Al-Dujaili, T. Schmiedlechner, E. Hemberg, and U.-M. O'Reilly, "Towards distributed coevolutionary GANs," in *AAAI Symposium*, 2018.

[7] J. Toutouh, E. Hemberg, and U.-M. O'Reilly, "Spatial evolutionary generative adversarial networks," in *Proceedings of the Genetic and Evolutionary Computation Conference*, ser. GECCO '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 472–480.

[8] T. Schmiedlechner, I. N. Z. Yong, A. Al-Dujaili, E. Hemberg, and U.-M. O'Reilly, "Lipizzaner: A System That Scales Robust Generative Adversarial Network Training," in *the 32nd Conference on Neural Information Processing Systems (NeurIPS 2018) Workshop on Systems for ML and Open Source Software*.

[9] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, G. Klambauer, and S. Hochreiter, "GANs trained by a two time-scale update rule converge to a nash equilibrium," *arXiv preprint arXiv:1706.08500*, vol. 12, no. 1, 2017.

[10] J. Toutouh, E. Hemberg, and U.-M. O'Reilly, "Data dieting in gan training," in *Deep Neural Evolution*. Springer, 2020, pp. 379–400.

[11] ——, "Analyzing the components of distributed coevolutionary gan training," in *International Conference on Parallel Problem Solving from Nature*. Springer, 2020, pp. 552–566.

Revisiting data complexity metrics based on morphology for overlap and imbalance: snapshot, new Overlap Number of Balls metrics and singular problems prospect

José Daniel Pascual-Triana, David Charte, Marta Andrés Arroyo, Alberto Fernández, Francisco Herrera
DaSCI Andalusian Institute of Data Science and Computational Intelligence
University of Granada
Granada, Spain
jdpascualt@ugr.es,{fdavidcl,alberto,herrera}@decsai.ugr.es

***Abstract*—This is a reduced version of our article published in Knowledge and Information Systems (KAIS) [1], presented to CAEPIA 20/21 KeyWorks. In said manuscript, novel Overlap Number of Balls complexity metrics, based on data morphology by using class-dependent open ball coverage of the instances, are presented. These metrics are compared to the state-of-the-art data complexity metrics, in terms of correlation with classification model performance and overlap estimation, obtaining good results in both fronts. Prospects are offered for the adaptation of these new metrics to singular problems.**

***Index Terms*—Data complexity metrics, Overlap, Morphology, Imbalanced classification, Singular problems**

I. Introduction

Supervised classification, where machine learning models are generated from pre-labeled data in order to classify new samples, has been used extensively to draw conclusions from data. Datasets, however, can often prove difficult to classify due to their inherent characteristics.

Data complexity metrics allow for the study of multiple properties of a dataset prior to learning a classification model [2]. They help grasp the characteristics that might negatively affect the learning process, which can indicate ways the data could be preprocessed in order to obtain models with better performance, or even which types of models are more suitable to learn from said dataset.

However, most data complexity metrics focus on a single property of the data (such as feature overlap, data sparsity or class balance), which might not be enough to properly evaluate the difficulty of the classification problem. For instance, class overlap (the presence of elements from multiple classes in the same area) can be hard to assess and usually proves detrimental for data classification, especially in the presence of class imbalance (some classes having significantly more elements than others) [3].

The aims of this work are fourfold: (1) to revisit data complexity metrics, and in particular those related to data morphology as a way to take into account multiple data characteristics simultaneously (such as class overlap, class imbalance and data neighbourhoods); (2) to propose new morphology metrics (the Overlap Number of Balls metrics, or ONB); (3) to analyse their capabilities at estimating classification performance and class overlap; and (4) to provide insight on how they can be applied on singular classification problems (those that involve more complex inputs, such as grouped instances, or outputs, such as multiple labels per instance).

The rest of the paper is structured as follows. Section II explains the methodology used in order to meet the four goals. Section III presents the conclusions reached.

II. Methodology

In order to revisit the state-of-the-art of data complexity metrics, the main articles in the topic were selected and, from them, a list of metrics was formed [2]. All those metrics were reflected and used in the paper, while those whose focus is more akin to morphology, the data neighbourhood metrics (N1, the Fraction of Borderline Points of the Minimum Spanning Tree; N2, the Ratio of Intra/Extra-Class Nearest Neighbour Distance; N3, the Error Rate of the Nearest Neighbour Classifier; N4, the Non-Linearity of the Nearest Neighbour Classifier; T1, the Fraction of Hyperspheres Covering Data; and LSC, the Local Set Average Cardinality), were explained in detail.

After explaining the state-of-the-art metrics, the new morphology metrics (ONB) were presented. They are based on the P-CCCD algorithm [4] for the coverage of data by class-dependent open balls. Four versions of the metric were introduced, depending on the type of distance used for the radii of the balls (Euclidean or Manhattan) and whether the number of balls is averaged by class.

This work has been partially supported by the Spanish Ministry of Economy and Competitiveness under project TIN2017-89517-P, including European Regional Development Funds, and the Andalusian regional project P18-FR-4961. This work is part of the PRII2018-02 Intensification Program from the University of Granada and the FPU National Program (Ref. FPU17/04069).

To study the capabilities of the proposed ONB morphology metrics for the estimation of class overlap and model performance, the correlations of the performance of K-Nearest Neighbours, Decision Tree and Naïve Bayes classifiers with the ONB and state-of-the-art metrics [2], over a base of artificial datasets (72 class-balanced and 585 class-imbalanced, generated to have controlled theoretical class overlap and imbalance) and 23 real datasets from KEEL, were contrasted. The correlations of the chosen metrics with class overlap were also studied for the artificial datasets.

Overall, ONB metrics achieved higher correlations with performance than the state-of-the-art metrics (see Table I for a sample, showing the correlations between performance of the chosen classifiers, measured using the geometric mean, and the overall winning metrics, N1, N3 and ONB averaged by class and using Manhattan distance radii, for the imbalanced artificial datasets) while also being competent at overlap estimation. This makes them suitable for assessing the complexity of classification problems a priori.

TABLE I
CORRELATIONS BETWEEN CLASSIFIER PERFORMANCE, MEASURED BY THE GEOMETRIC MEAN, AND THE OVERALL WINNING METRICS FOR THE 585 IMBALANCED ARTIFICIAL DATASETS.

Classifier	Correlation with performance		
	N3	N1	ONB^{man}_{avg}
1NN	-0.64221	-0.63954	**-0.84805**
3NN	-0.57571	-0.57708	**-0.79894**
DT	-0.55215	-0.56526	**-0.74016**
NB	-0.49918	-0.56023	**-0.64383**

Having shown the capabilities of the ONB metrics, this work also provided prospects for their adaptation to singular problems [5], where the use of data complexity metrics was often incompatible due to the change in data structures. Following these prospects should allow for the use of the ONB metrics to provide further insight on singular problems.

III. CONCLUSIONS

This manuscript has revisited data complexity metrics, focusing on data morphology as a way to estimate classification performance and class overlap. New ONB morphology metrics were proposed and their capabilities were analysed and compared to other state-of-the-art complexity metrics, showing promise in both estimations. Then, insight was provided on how these new metrics can be applied on singular classification problems, where usual complexity metrics are usually incompatible.

REFERENCES

[1] J. D. Pascual-Triana, D. Charte, M. Andrés Arroyo, A. Fernández, and F. Herrera, "Revisiting data complexity metrics based on morphology for overlap and imbalance: snapshot, new overlap number of balls metrics and singular problems prospect," *Knowledge and Information Systems*, vol. 63, p. 1961–1989, 2021, DOI: 10.1007/s10115-021-01577-1.

[2] A. C. Lorena, L. P. F. Garcia, J. Lehmann, M. C. P. Souto, and T. K. Ho, "How complex is your classification problem? a survey on measuring classification complexity," *ACM Comput. Surv.*, vol. 52, no. 5, 2019.

[3] A. Fernández, S. García, M. Galar, R. Prati, B. Krawczyk, and F. Herrera, *Learning from Imbalanced Data Sets*. Springer, 2018.

[4] A. Manukyan and E. Ceyhan, "Classification of Imbalanced Data with a Geometric Digraph Family," *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 6504–6543, 2016.

[5] D. Charte, F. Charte, S. García, and F. Herrera, "A snapshot on non-standard supervised learning problems: taxonomy, relationships, problem transformations and algorithm adaptations," *Progress in Artificial Intelligence*, vol. 8, no. 1, pp. 1–14, 2019.

XX Congreso Español sobre Tecnologías y Lógica Fuzzy



XX Congreso Español sobre Tecnologías y Lógica Fuzzy

Sesión General



Multiple fuzzy Sugeno λ-measures in networks.

1st Inmaculada Gutiérrez
Facultad de Estudios Estadísticos
Universidad Complutense de Madrid
Madrid, España
inmaguti@ucm.es
ORCID: 0000-0002-7103-393X

2nd Daniel Gómez
Facultad de Estudios Estadísticos
Instituto de Evaluación Sanitaria
Universidad Complutense de Madrid
Madrid, España
dagomezc@estad.ucm.es
ORCID: 0000-0001-9548-5781

3rd Javier Castro
Facultad de Estudios Estadísticos
Instituto de Evaluación Sanitaria
Universidad Complutense de Madrid
Madrid, España
jcastroc@estad.ucm.es
ORCID: 0000-0002-5345-8870

4th Rosa Espínola
Facultad de Estudios Estadísticos
Instituto de Evaluación Sanitaria
Universidad Complutense de Madrid
Madrid, España
rosaev@estad.ucm.es
ORCID: 0000-0003-2336-3659

***Abstract*—In this paper we take up the community detection problem based on fuzzy measures. We focus on the existence of a family of vectors which define some additional information about the individuals of a network, from which we obtain multiple fuzzy Sugeno λ-measures. We introduce a new knowledge representation model, which combines the information of those fuzzy measures with a crisp graph: the multi-dimensional extended vector fuzzy graph. We suggest a particular application of it, devoted to the community detection problem. To solve it, we define a method, based on the consideration of the multi-dimensional weighted graph associated with multiple fuzzy measures.**

***Index Terms*—Fuzzy measure, Sugeno λ-measure, Community detection problem, Extended vector fuzzy graph, Weighted graph associated with a fuzzy measure**

I. Introduction

The community detection problem is one of the most important topics in the field of Social Networks Analytics (SNA). Classical methods have their basis on the structure of the graph, an assumption which has provided good results. Nevertheless, in the last years, some authors have agreed on the importance of adding some additional information apart from the graph when dealing with SNA problems, particularly, when finding communities in a network. Different approaches can be found. Particularly, Gutierrez et al. have been working in the incorporation of some additional information modeled by fuzzy measures to enrich the process [1]–[3].

We take up the idea introduced in [4] of finding communities by considering the additional information modeled by a vector, for example, the ratings of a film given by a group of people connected among them. Several synergies among individuals may appear from the ratings, modeled by a fuzzy Sugeno $\lambda-$measure [5], [6], so the community structure may be affected. Draw from this initial assumption, now we suggest the management of multiple vectors. For example, imagine each vector represents the interest of a group of people regarding an activity, so that one vector is about sports, another about music or literature, and so on. From these interests, there may appear different synergies among the people, which will be represented by a multi-dimensional family of fuzzy Sugeno λ-measures, obtained from mentioned vectors. To handle these fuzzy Sugeno λ-measures, it is defined the multi-dimensional weighted graph associated with them (vector MAGW). It represents the synergies among individuals regarding the information given by the vectors. This tool has a many application in any SNA problem. Particularly, we will use it to address a community detection problem with additional information.

On the other hand, as a generalization of the extended vector fuzzy graph [4], we define the multi-dimensional extended vector fuzzy graph (MEVFG), which combines the knowledge of a crisp graph with that of multiple fuzzy Sugeno λ-measures obtained from vectors. On its basis, we approach a community detection problem based on fuzzy measures. We define an algorithm inspired by the Louvain method. It is based on modularity optimization [7] and local moving [8]. We also propose a specification of it to deal with 1-additive measures, situation in which the proposed algorithm is polynomial-time complexity.

The remainder of the paper is organized as follows. In Section II we provide several useful definitions which will be used later. Then, in Section III we address an scenario in which there are multiple fuzzy measures defining some information about a graph. The starting point is the existence of a family of vectors, each of which defines an evidence about the

This research has been partially supported by the Government of Spain, Grant Plan Nacional de I+D+i, MTM2015-70550-P, PGC2018096509-B-I00, BDNS Identif.: 498817 EB25/20 and CT17/17 - CT18/17.

individuals of the graph, i.e. the nodes. After that, we propose a particular application of the multi-dimensional extended vector fuzzy graph devoted to the community detection in Section IV. We conclude the work in Section V with some conclusions and future research lines.

II. Preliminaries

In this section we show several definitions which set the basis of this paper. We start with the definition of a graph or network, a tool which set the basis of this work.

Definition 1. ***Graph / Network [9].***

A graph or network is a pair $G = (V, E)$ where $V = \{1, 2, \ldots, n\}$ is a set of individuals named nodes or vertices, and $E = \{\{i, j\} \mid i, j \in V\}$ is a non ordered set of pairs of nodes, named edges or arcs.

Another way to characterize a graph is by means of its adjacency matrix. This matrix, usually denoted by A, represents the direct connections between the nodes, in the sense that $A_{ij} = 1$ if $\{i, j\} \in E$, and $A_{ij} = 0$, otherwise.

On the other hand, we will consider non-weighted graphs, (and edge exists or not, so $A_{ij} = 1$ or $A_{ij} = 0$), and weighted graphs. In this case, there is a weight function defined on the set of edges, $w : E \to \mathbb{R}$, so there is a weight or value assigned to each edge. In this scenario, $A_{ij} = w_{ij}$, where $w_i j$ is the weight of the edge $\{i, j\} \in E$.

Besides the graphs, the cornerstone of this paper is the use of fuzzy Sugeno λ-measures [4], functions to which we force to be fuzzy measures [6] and Sugeno λ-measures [5]. Particularly, we focus on this type of functions when they are obtained from a vector.

Definition 2. ***Fuzzy Sugeno λ-measure obtained from a vector ($\mu_{\mathbf{x},\mathbf{p}}$) [4].***

Let $x = (x_1, \ldots, x_n)$ denote a vector defining any evidence about the elements of the n-set V, where $x_i \geq 0$ $\forall i$. Let $p \in (0, 1]$ denote a parameter. The function $\mu_{x,p}$ is fuzzy Sugeno λ-measure obtained from a vector, and it is characterized as follows:

$$\mu_{x,p}(i) = \frac{p x_i}{\sum_{k=1}^{n} x_k}, \forall i \in V$$

and

$$\mu_{x,p}(A \cup B) = \mu_{x,p}(A) + \mu_{x,p}(B) + \lambda \mu_{x,p}(A)\mu_{x,p}(B),$$
$$\forall A, B \subseteq V, \text{ with } A \cap B = \emptyset \text{ and}$$
$$\lambda + 1 = \prod_{i=1}^{n}(1 + \lambda \mu_{x,p}(i))$$

To simplify the visualization and understanding of the relations defined by $\mu_{x,p}$, we will suggest the definition of the weighted graph associated with it. To characterize this graph, we consider the Shapley value. It is an essential tool in the frame of Game Theory which was also adapted to the fuzzy measures background.

Definition 3. ***Shapley value [10].***

Let $\mu : 2^V \to [0, 1]$ denote a fuzzy measure, where $|\mathbf{V}| = n$. For every $i \in V$ its Shapley index is calculated as:

$$Sh_i(\mu) = \sum_{K \subseteq V \setminus \{i\}} \frac{(n - |K| - 1)!|K|!}{n!} (\mu(K \cup \{i\}) - \mu(K))$$

The Shapley value of the fuzzy measure $\mu : \mathcal{P}(V) \to [0, 1]$ is defined by the vector $Sh(\mu) = (Sh_1(\mu), \ldots, Sh_n(\mu))$.

Hence, on the basis of the Shapely value and considering an aggregation operator, we recall the concept of weighted graph associated with a fuzzy measure [11], adapted for the scenario in which the fuzzy measure is obtained from a vector [4]. This graph is a tool which represents the synergies and relations between every pair of elements of the set V, according to the knowledge modeled by $\mu_{x,p}$.

Definition 4. ***Weighted graph associated with $\mu_{\mathbf{x},\mathbf{p}}, G_{\mu_{x,p}}$ (AWG of vector) [4].***

Let V denote a n-set, and let x denote a n-vector defining any evidence about the elements of V. Let $\mu_{x,p}$ denote the fuzzy Sugeno λ-measure obtained from x. The weighted graph associated with $\mu_{x,p}$, $G_{\mu_{x,p}}$, (AWG of vector) is that whose adjacency matrix is:

$$X_{ij} = \phi\left(Sh_i(\mu_{x,p}) - Sh_i^j(\mu_{x,p}), Sh_j(\mu_{x,p}) - Sh_j^i(\mu_{x,p})\right) \quad (1)$$

being $\phi : [-1, 1]^2 \to [0, 1]$ a bi-variate aggregation operator [12]; $Sh_i(\mu_{x,p})$ and $Sh_i^j(\mu_{x,p})$ the Shapley values of i on μ_x when it is in a coalition with all the elements of V or $V \setminus \{j\}$, respectively [10].

For every pair of elements of V, the AWG represents how each individual is affected by the absence of the other in a coalition regarding $\mu_{x,p}$.

The calculation of the Shapely value may be hard for general fuzzy measures; nevertheless it is much easier for additive fuzzy measures. In [4] it was demonstrated that $\mu_{x,p}$ is a fuzzy Sugeno λ-measure. Particularly, when $p = 1$, it is 1-additive [13] (denoted by μ_x^a). To facilitate the calculation of the Shapley value when defining the weighted graph associated with $\mu_{x,p}$, we focus on this scenario.

To end this section, we present another tool which plays an essential role to address the community detection problem based on fuzzy measures is the extended fuzzy graph, firstly introduced in [11] and then adapted to the particular case of fuzzy Sugeno λ-measures in [4].

Definition 5. ***Extended vector fuzzy graph (EVFG) [4].***

Let $G = (V, E)$ denote a crisp graph and let x denote a vector, so that $\mu_{x,p}$ is the fuzzy Sugeno λ-measure obtained from x. The triplet $\widetilde{G}_x = (V, E, \mu_{x,p})$ is an extended vector fuzzy graph, (EVFG).

III. Multiple fuzzy Sugeno λ- measures in a graph

In the framework of networks analysis, we assume the existence of several information sources about the nodes of the graphs, given by a family of vectors, $\left(x^1, \ldots, x^r\right)$, so that each of them defines (independently) an evidence about the individuals of a set. From these family of vectors, we characterize the fuzzy Sugeno λ-measures [4] $\mu_{x^1,p^1}, \ldots, \mu_{x^r,p^r}$.

For a proper understanding and visualization of these fuzzy measures, we suggest the definition of the corresponding multi-dimensional weighted graph associated with them. It is a generalization to a multi-dimensional scale of the weighted graph associated with a fuzzy Sugeno λ-measure obtained from a vector (AWG of vector).

Definition 6. ***Multi-dimensional weighted graph associated with a family of fuzzy Sugeno λ-measures (vector MAWG).***

Let $\left(x^1, \ldots, x^r\right)$ denote a family of vectors; each one defines an evidence about the elements of a set V with $|V| = n$. $\left(\mu_{x^1,p^1}, \ldots, \mu_{x^r,p^r}\right)$ denote the fuzzy Sugeno λ-measures obtained from these vectors. The multi-dimensional weighted graph associated with $\left(\mu_{x^1,p^1}, \ldots, \mu_{x^r,p^r}\right)$ is that whose adjacency matrices are $\left(X^1, \ldots, X^r\right)$, being X^ℓ the adjacency matrix of the AWG of μ_{x^ℓ,p^ℓ}, where, $\forall \ell = 1, \ldots, r$, $\forall i, j \in V$,

$$X^\ell_{ij} = \phi^\ell \left(Sh_i(\mu_{x^\ell,p^\ell}) - Sh^j_i(\mu_{x^\ell,p^\ell}), Sh_j(\mu_{x^\ell,p^\ell}) - Sh^i_j(\mu_{x^\ell,p^\ell}) \right) \quad (2)$$

being $\phi : [-1, 1]^2 \to [0, 1]$ a bi-variate aggregation operator [12]; $Sh_i(\mu_{x,p})$ and $Sh^j_i(\mu_{x,p})$ the Shapley indices of i on μ_x when it is in a coalition with all the elements of V or $V \backslash \{j\}$, respectively [10].

Then, we define a knowledge representation model which combines the information provided by a crisp graph with some additional information independent of its structure which is modeled by some vectors.

Definition 7. ***Multi-dimensional extended vector fuzzy graph (MEVFG).***

Let $G = (V, E)$ denote a crisp graph, and let $\left(x^1, \ldots, x^r\right)$ denote a family of vectors so that each of them defines any evidence about the elements of V. Let $\left(\mu_{x^1,p^1}, \ldots, \mu_{x^r,p^r}\right)$ denote the family of fuzzy Sugeno λ-measures obtained from vectors $\left(x^1, \ldots, x^r\right)$. Then, $\widetilde{G} = \left(V, E\left(\mu_{x^1,p^1}, \ldots, \mu_{x^r,p^r}\right)\right)$ is a multi-dimensional extended vector fuzzy graph, (MEVFG).

Let us note that the MEVFG goes further than other existent tool: it allows the characterization of several synergies among the individuals, regardless their connections in the crisp graph. Furthermore, due to the properties of the fuzzy Sugeno λ-measures, some relations between elements can be inferred from the knowledge about some individual evidence.

IV. Community detection problem in a multi-dimensional extended vector fuzzy graph

The MEVFG has numerous applications. In this paper we take up the idea introduced in [4] related to community detection in graphs according to the existence of a fuzzy Sugeno λ-measure. Now we work with the multi-dimensional extended vector fuzzy graph $\widetilde{G} = \left(V, E, \left(\mu_{x^1,p^1}, \ldots, \mu_{x^r,p^r}\right)\right)$ obtained from the combination of a crisp graph $G = (V, E)$ and a family of vectors $\left(x^1, \ldots, x^r\right)$ defining additional information about the elements of V, from which we define the fuzzy Sugeno $\lambda-$measures $\left(\mu_{x^1,p^1}, \ldots, \mu_{x^r,p^r}\right)$. We agree that, the more information is analyzed, the more cohesive and realistic are the groups detected.

The proposed methodology, named *Multi-dimensional Sugeno Louvain*, is based on the Louvain Algorithm [14]. The main point of our algorithm is to summarize all the knowledge of the MEVFG into two matrices: the adjacency matrix of the crisp graph, A, represents the direct connections between the nodes (edges), and $\mathbb{X}$ summarizes the additional information given by the family of vectors $\left(x^1, \ldots, x^r\right)$.

- **Step 1: definition of the vector MAWG.** Given the fuzzy Sugeno λ-measures $\left(\mu_{x^1,p^1}, \ldots, \mu_{x^r,p^r}\right)$ obtained from $\left(x^1, \ldots x^r\right)$, matrices $\left(X^1, \ldots, X^r\right)$ have to be defined according to equation (2).

- **Step 2: information aggregation.** Matrices $X^1, \ldots, X^r$ are aggregated to obtain the matrix $\mathbb{X}$. The aggregation function $\Phi : \Pi^r \to \Pi$ is used, being Π the set of quadratic n-matrices. Particularly, we suggest the use of a matrix aggregator based on the classical aggregation operators with "element to element" transformation: $\mathbb{X} = \Phi\left(X^1, \ldots, X^r\right)$.

After this aggregation process, the method *Duo Louvain*, summarized by its pseudo-code in the Algorithm 1, has to be applied (see [15], [16] for more details), considering the matrix $M = \theta(A, \mathbb{X})$, being $\theta : \Pi^2 \to \Pi$ an aggregation function. That method can consider the information of two matrices when finding communities in a graph (A is used to find "feasible" communities, and any other matrix to calculate the maximum of modularity). Note that the notion of group and its size depend on the operator Φ considered [16]. The new community detection method defined to find groups in a multi-dimensional extended vector fuzzy graph is summarized in the Algorithm 2 and it is named *Multi-dimensional Sugeno-Louvain*.

Algorithm 1 *Duo Louvain*

Input: $(A,\ M)$;
Output: P;
Preliminary
$C_i \leftarrow \{i\}, \forall i \in V$ (ech node i is an isolated community);
$P \leftarrow (1, 2, \ldots, n)$ (initial partition);
end Preliminary
Phase 1
Take $o = \left(o^1, \ldots, o^i, \ldots, o^n\right) \in \pi(V)$;
$stop \leftarrow 0$;
while $(stop == 0)$ **do**
  $stop \leftarrow 1$
  for $(i = 1)$ **to** (n) **do**
    $(e_1, \ldots, e_h) \leftarrow H(o^i)$ (find the neighbours of o^i in A);
    for $(j = 1)$ **to** (h) **do**
      Calculate $\Delta Q_{o^i}(e_j)$ in M;
    end for
    $j^* \leftarrow \left\{e_\ell \mid \Delta Q_{o^i}(j^*) = \max_{\ell \in \{1\ldots,h\}} \left\{\Delta Q_{o^i}(e_\ell)\right\}\right\}$;
    if $(\Delta Q_{o^i}(j^*) > 0)$ **then**
      $C_{P(o^i)} \leftarrow C_{P(o^i)} \backslash \{o^i\}$;
      $C_{P(j^*)} \leftarrow C_{P(j^*)} \cup \{o^i\}$;
      $P\left(o^i\right) \leftarrow P\left(j^*\right)$;
      $stop \leftarrow 0$;
    end if
  end for
end while
end Phase 1
Phase 2
Calculate A^* from A (nodes of A^* are the communities previously found in A);
Calculate M^* from M (nodes of M^* are the communities previously found in M);
if $(A^* \neq A)$ **then**
  $A \leftarrow A^*$;
  $M \leftarrow M^*$;
  Apply **Phase 1** and **Phase 2**;
end if
end Phase 2
return(P);

Algorithm 2 *Multi-dimensional Sugeno-Louvain*

Input: $\left(A, \left(\ x^1, \ldots, x^r\right), \left(\ p^1, \ldots, p^r\right)\right)$, A represents $G = (V, E)$;
Output: P;
Preliminary
for $(\ell = 1)$ **to** (r) **do**
  Calculate μ_{x^ℓ, p^ℓ} (fuzzy Sugeno λ-measure from x^ℓ);
  $X^\ell_{ij} \leftarrow \phi^\ell \left(Sh_i(\mu_{x^\ell,p^\ell}) - Sh^j_i(\mu_{x^\ell,p^\ell})Sh_j(\mu_{x^\ell,p^\ell}) - Sh^i_j(\mu_{x^\ell,p^\ell})\right)$, $\forall i, j \in V$;
end for
$\mathbb{X} \leftarrow \Phi\left(X^1, \ldots, X^r\right)$;
$M \leftarrow \theta\left(A, \mathbb{X}\right)$;
end Preliminary
$P \leftarrow$ *Duo Louvain*(A, M);
return(P);

The exponential complexity concerning the Shapley value may be avoid by considering additive fuzzy measure. Then, we suggest a specific application of the *Multi-dimensional Sugeno-Louvain* Algorithm, named *1-additive Multi-dimensional Sugeno Louvain*, which involves a particular 1-additive characterization of μ_{x^ℓ,p^ℓ}, denoted by $\mu^a_{x^\ell}$. On this basis, the calculation of the Shapley index is immediate from vector x^ℓ as follows: $Sh_i(\mu^a_{x^\ell}) = \frac{{x^\ell}_i}{\sum_{k=1}^n {x^\ell}_k}$ and $Sh^j_i(\mu^a_{x^\ell}) = \frac{{x^\ell}_i}{\sum_{\substack{k=1 \\ k\neq j}}^n {x^\ell}_k}$. Then, the complexity of the method *1-additive Multi-dimensional Sugeno-Louvain* (its pseudo-code is showed in Algorithm 3), whose only difference with respect to the *Multidimensional-dimensional Sugeno-Louvain* is the calculation of X^ℓ (line *6) in pseudo-code* is equal to the Louvain Algorithm.

Algorithm 3 *1-additive Multi-dimensional Sugeno Louvain*

Input: $\left(A,\ \left(x^1, \ldots, x^r\right)\right)$, A is a representation of $G = (V, E)$;
Output: P;
Preliminary
for $(\ell = 1)$ **to** (r) **do**
  $X^\ell_{ij} \leftarrow \phi\{|\frac{x^\ell_i}{\sum_{k=1}^n x^\ell_k} - \frac{x^\ell_i}{\sum_{\substack{k=1 \\ k\neq j}}^n x^\ell_k}|,\ |\frac{x^\ell_j}{\sum_{k=1}^n x^\ell_k} - \frac{x^\ell_j}{\sum_{\substack{k=1 \\ k\neq i}}^n x^\ell_k}|\}$;
end for
$\mathbb{X} \leftarrow \Phi\left(X^1, \ldots, X^r\right)$;
$M \leftarrow \theta\left(A, \mathbb{X}\right)$;
end Preliminary
$P \leftarrow$ *Duo Louvain*(A, M);
return(P);

Example 1.

We consider graph $G = (V, E)$ whose nodes are connected as a chain of size 12, as it can be seen in matrix A.

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

Fig. 1. Adjacency matrix of graph $G = (V, E)$

There exist 4 vectors of additional information, defining some evidences about the elements of V,

- $x^1 = (\mathbf{9}, \mathbf{9.5}, \mathbf{10}, 1, 0.5, 1, \mathbf{9.5}, \mathbf{8}, \mathbf{10}, 1, 1.5, 1)$
- $x^2 = (\mathbf{10}, \mathbf{9.5}, \mathbf{9}, 1, 0.5, 1, \mathbf{9}, \mathbf{9}, \mathbf{9.5}, 1.5, 2, 0.5)$
- $x^3 = (\mathbf{9.5}, \mathbf{8.5}, \mathbf{10}, 1.5, 1, 1, \mathbf{10}, \mathbf{9.5}, \mathbf{9.5}, 0.9, 1, 1)$
- $x^4 = (\mathbf{9}, \mathbf{9.5}, \mathbf{10}, 1, 1, 1, \mathbf{10}, \mathbf{9.5}, \mathbf{9}, 0.5, 1, 1)$

Each vector represents the opinion (ratings) of 12 people about a films.

From vectors $\left(x^1, x^2, x^3, x^4\right)$, we define the family of $1-$additive fuzzy Sugeno $\lambda-$measures $\left(\mu^a_{x^1}, \mu^a_{x^2}, \mu^a_{x^3}, \mu^a_{x^4}\right)$. The adjacency matrices of the corresponding vector MAWG are (X^1, X^2, X^3, X^4), showed in the Figure 2.

We accept that there are more synergies between those people who have similar preferences. Any classical community detection algorithm based on modularity optimization provides the partition $P = \{\{1, 2, 3, 4\}, \{5, 6, 7, 8\}, \{9, 10, 11, 12\}\}$. Nevertheless, if the additional information is considered, the partition provided by the Multi-dimensional Sugeno-Louvain 1-additive is $P^x = \{\{1, 2, 3\}, \{4, 5, 6\}, \{7, 8, 9\}, \{10, 11, 12\}\}$, which respect the synergies between the elements (showed in matrices X^1, X^2, X^3, X^4, whose highest values are bold), as well as the structure of the crisp graph established by the edges.

$$X^1 = \frac{1}{10000}\begin{pmatrix} 0 & \mathbf{260} & \mathbf{274} & 24 & 12 & 24 & \mathbf{260} & \mathbf{215} & \mathbf{274} & 24 & 36 & 24 \\ \mathbf{260} & 0 & \mathbf{292} & 25 & 12 & 25 & \mathbf{277} & \mathbf{227} & \mathbf{292} & 25 & 38 & 25 \\ \mathbf{274} & \mathbf{292} & 0 & 26 & 13 & 26 & \mathbf{292} & \mathbf{239} & \mathbf{310} & 26 & 40 & 26 \\ 24 & 25 & 26 & 0 & 1 & 3 & 25 & 21 & 26 & 3 & 4 & 3 \\ 12 & 12 & 13 & 1 & 0 & 1 & 12 & 10 & 13 & 1 & 2 & 1 \\ 24 & 25 & 26 & 3 & 1 & 0 & 25 & 21 & 26 & 3 & 4 & 3 \\ \mathbf{260} & \mathbf{277} & \mathbf{292} & 25 & 12 & 25 & 0 & \mathbf{227} & \mathbf{292} & 25 & 38 & 25 \\ \mathbf{215} & \mathbf{227} & \mathbf{239} & 21 & 10 & 21 & \mathbf{227} & 0 & \mathbf{239} & 21 & 32 & 21 \\ \mathbf{274} & \mathbf{292} & \mathbf{310} & 26 & 13 & 26 & \mathbf{292} & \mathbf{239} & 0 & 26 & 40 & 26 \\ 24 & 25 & 26 & 3 & 1 & 3 & 25 & 21 & 26 & 0 & 4 & 3 \\ 36 & 38 & 40 & 4 & 2 & 4 & 38 & 32 & 40 & 4 & 0 & 4 \\ 24 & 25 & 26 & 3 & 1 & 3 & 25 & 21 & 26 & 3 & 4 & 0 \end{pmatrix}$$

$$X^2 = \frac{1}{10000}\begin{pmatrix} 0 & \mathbf{287} & \mathbf{269} & 26 & 13 & 26 & \mathbf{269} & \mathbf{269} & \mathbf{287} & 39 & 53 & 13 \\ \mathbf{287} & 0 & \mathbf{256} & 25 & 12 & 25 & \mathbf{256} & \mathbf{256} & \mathbf{272} & 37 & 50 & 12 \\ \mathbf{269} & \mathbf{256} & 0 & 23 & 12 & 23 & \mathbf{242} & \mathbf{242} & \mathbf{256} & 35 & 48 & 12 \\ 26 & 25 & 23 & 0 & 1 & 3 & 23 & 23 & 25 & 4 & 5 & 1 \\ 13 & 12 & 12 & 1 & 0 & 1 & 12 & 12 & 12 & 2 & 3 & 1 \\ 26 & 25 & 23 & 3 & 1 & 0 & 23 & 23 & 25 & 4 & 5 & 1 \\ \mathbf{269} & \mathbf{256} & \mathbf{242} & 23 & 12 & 23 & 0 & \mathbf{242} & \mathbf{256} & 35 & 48 & 12 \\ \mathbf{269} & \mathbf{256} & \mathbf{242} & 23 & 12 & 23 & \mathbf{242} & 0 & \mathbf{256} & 35 & 48 & 12 \\ \mathbf{287} & \mathbf{272} & \mathbf{256} & 25 & 12 & 25 & \mathbf{256} & \mathbf{256} & 0 & 37 & 50 & 12 \\ 39 & 37 & 35 & 4 & 2 & 4 & 35 & 35 & 37 & 0 & 8 & 2 \\ 53 & 50 & 48 & 5 & 3 & 5 & 48 & 48 & 50 & 8 & 0 & 3 \\ 13 & 12 & 12 & 1 & 1 & 1 & 12 & 12 & 12 & 2 & 3 & 0 \end{pmatrix}$$

$$X^3 = \frac{1}{10000}\begin{pmatrix} 0 & \mathbf{232} & \mathbf{278} & 36 & 24 & 24 & \mathbf{278} & \mathbf{264} & \mathbf{264} & 22 & 24 & 24 \\ \mathbf{232} & 0 & \mathbf{244} & 32 & 21 & 21 & \mathbf{244} & \mathbf{232} & \mathbf{232} & 19 & 21 & 21 \\ \mathbf{278} & \mathbf{244} & 0 & 38 & 25 & 25 & \mathbf{295} & \mathbf{278} & \mathbf{278} & 23 & 25 & 25 \\ 36 & 32 & 38 & 0 & 4 & 4 & 38 & 36 & 36 & 3 & 4 & 4 \\ 24 & 21 & 25 & 4 & 0 & 3 & 25 & 24 & 24 & 2 & 3 & 3 \\ 24 & 21 & 25 & 4 & 3 & 0 & 25 & 24 & 24 & 2 & 3 & 3 \\ \mathbf{278} & \mathbf{244} & \mathbf{295} & 38 & 25 & 25 & 0 & \mathbf{278} & \mathbf{278} & 23 & 25 & 25 \\ \mathbf{264} & \mathbf{232} & \mathbf{278} & 36 & 24 & 24 & \mathbf{278} & 0 & \mathbf{264} & 22 & 24 & 24 \\ \mathbf{264} & \mathbf{232} & \mathbf{278} & 36 & 24 & 24 & \mathbf{278} & \mathbf{264} & 0 & 22 & 24 & 24 \\ 22 & 19 & 23 & 3 & 2 & 2 & 23 & 22 & 22 & 0 & 2 & 2 \\ 24 & 21 & 25 & 4 & 3 & 3 & 25 & 24 & 24 & 2 & 0 & 3 \\ 24 & 21 & 25 & 4 & 3 & 3 & 25 & 24 & 24 & 2 & 3 & 0 \end{pmatrix}$$

$$X^4 = \frac{1}{10000}\begin{pmatrix} 0 & \mathbf{256} & \mathbf{269} & 23 & 23 & 23 & \mathbf{269} & \mathbf{256} & \mathbf{242} & 12 & 23 & 23 \\ \mathbf{256} & 0 & \mathbf{287} & 25 & 25 & 25 & \mathbf{287} & \mathbf{272} & \mathbf{256} & 12 & 25 & 25 \\ \mathbf{269} & \mathbf{287} & 0 & 26 & 26 & 26 & \mathbf{305} & \mathbf{287} & \mathbf{269} & 13 & 26 & 26 \\ 23 & 25 & 26 & 0 & 3 & 3 & 26 & 25 & 23 & 1 & 3 & 3 \\ 23 & 25 & 26 & 3 & 0 & 3 & 26 & 25 & 23 & 1 & 3 & 3 \\ 23 & 25 & 26 & 3 & 3 & 0 & 26 & 25 & 23 & 1 & 3 & 3 \\ \mathbf{269} & \mathbf{287} & \mathbf{305} & 26 & 26 & 26 & 0 & \mathbf{287} & \mathbf{269} & 13 & 26 & 26 \\ \mathbf{256} & \mathbf{272} & \mathbf{287} & 25 & 25 & 25 & \mathbf{287} & 0 & \mathbf{256} & 12 & 25 & 25 \\ \mathbf{242} & \mathbf{256} & \mathbf{269} & 23 & 23 & 23 & \mathbf{269} & \mathbf{256} & 0 & 12 & 23 & 23 \\ 12 & 12 & 13 & 1 & 1 & 1 & 13 & 12 & 12 & 0 & 1 & 1 \\ 23 & 25 & 26 & 3 & 3 & 3 & 26 & 25 & 23 & 1 & 0 & 3 \\ 23 & 25 & 26 & 3 & 3 & 3 & 26 & 25 & 23 & 1 & 3 & 0 \end{pmatrix}$$

Fig. 2. Adjacency of the vector MAGW of $(\mu^a_{x^1}, \mu^a_{x^2}, \mu^a_{x^3}, \mu^a_{x^4})$

V. Conclusions

In this paper we address a new perception of the community detection problem in networks. In a multi-dimensional scale, we suggest the inclusion of some additional information defined by a family of vectors to the process of finding groups in a crisp graph. We assert that, the more information is analyzed, the more realistic the results obtained.

In the particular context of fuzzy Sugeno λ-measures, we introduce two new tools which facilitates the handling of a family of them. The first one is the multi-dimensional weighted graph associated with multiple fuzzy Sugeno λ-measures, a representation tool which shows the synergies between every

pair of elements regarding the knowledge modeled by the measures considered.

On the other hand, for an scenario modeled by a crisp graph $G = (V, E)$, about whose nodes there is some additional information defined by vectors $\left(x^1, \ldots, x^r\right)$, we define the multi-dimensional extended vector fuzzy graph $\widetilde{G} = \left(V, E, \left(\mu_{x^1,p^1}, \ldots, \mu_{x^r,p^r}\right)\right)$, being $\left(\mu_{x^1,p^1}, \ldots, \mu_{x^r,p^r}\right)$ the fuzzy Sugeno λ-measures obtained from those vectors [4].

On the basis of this tool, we approach the community detection problem based on multiple fuzzy measures. We define a new algorithm, *Multi-dimensional Sugeno Louvain*, which has two main points: (1) an aggregation process to summarize all the knowledge modeled by $\widetilde{G}$ into two matrices; (2) the application of the *Duo Louvain* Algorithm. We suggest a particular application of that algorithm to use 1-additive fuzzy measures, so that the calculation of the Shapley value is immediate. So, we can affirm that the complexity of the *1-additive Multi-dimensional Sugeno Louvain* Algorithm is equal to the Louvain Algorithm.

At the moment, we are currently working on the evaluation and testing of the proposed methodology. To carry on with it, we will differentiate two different steps. The first one, will be devoted to the consideration of synthetic networks. We will apply our algorithm in several benchmark models [17], then we will consider the Normalized Mutual Information (NMI) [18] to evaluate the results obtained. After that, we will work with some real cases. We are particularly interested in analyzing the data obtained from Social Networks, such as Twitter or Facebook.

The testing process is still in early stage, but the preliminary results we have obtained are very promising on the goodness of the proposed method.

REFERENCES

[1] M. Barroso, I. Gutiérrez, D. Gómez, J. Castro, and R. Espínola, "Group definition based on flow in community detection," in *Information Processing and Management of Uncertainty in Knowledge-Based Systems*. Springer, 2020, pp. 524–538.

[2] I. Gutiérrez, J. A. Guevara, D. Gómez, J. Castro, and R. Espínola, "Community detection problem based on polarization measures. an application to Twitter: the COVID-19 case in Spain," vol. 9, no. 4, 2021.

[3] I. Gutiérrez, M. Barroso, D. Gómez, J. Castro, and R. Espínola, "Pattern-based clustering problem based on fuzzy measures," *Developments of Artificial Intelligence Technologies in Computation and Robotics*, vol. 12, pp. 412–420, 2020.

[4] I. Gutiérrez, D. Gómez, J. Castro, and R. Espínola, "Fuzzy sugeno λ-measures and theirs applications to community detection problems." Glasgow, UK: IEEE International Conference on Fuzzy Systems, 2020, pp. 1–6.

[5] M. Sugeno, "Theory of fuzzy integrals and its applications," Ph.D. dissertation, Tokyo Institute of Technology, 1974.

[6] ——, "Fuzzy measures and fuzzy integrals: A survey," *Fuzzy Automata and Decision Processes*, vol. 78, pp. 89–102, 1977.

[7] M. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Physical Review E*, vol. 69, no. 026113, 2004.

[8] L. Waltman and N. J. Van Eck, "A smart local moving algorithm for large-scale modularity-based community detection," *Europ. Phys. Journal B*, vol. 86, no. 11, 2013.

[9] J. Sylvester, "Chemistry and algebra," *Nature 17*, vol. 3, no. 284, pp. 103–104, 1878.

[10] L. S. Shapley, "A value for n-person games," *Contribute. Theory Games*, vol. 2, no. 28, pp. 307–317, 1953.

[11] I. Gutiérrez, D. Gómez, J. Castro, and R. Espínola, "A new community detection algorithm based on fuzzy measures," in *Advances in Intelligent Systems and Computing series*, vol. 1029, Intelligent and Fuzzy Techniques in Big Data Analytics and Decision Making. INFUS 2019. Springer, Cham, 2020, pp. 133–140.

[12] R. R. Yager and J. Kacprzyk, Eds., *The Ordered Weighted Averaging Operators. Theory and Applications*. Boston, MA, USA: Springer Science & Business Media, 1997.

[13] M. Grabisch, "k-order additive discrete fuzzy measures and their representation," *Fuzzy Sets Systems*, vol. 92, no. 2, pp. 167–189, 1997.

[14] V. Blondel, J. Guillaume, R. Lambiotte, and E. Lefevre, "Fast unfolding of communities in large networks," *Journal Of Statistical Mechanics-Theory And Experiment*, vol. 10, no. P10008, 2008.

[15] I. Gutiérrez, D. Gómez, J. Castro, and R. Espínola, "Fuzzy measures: A solution to deal with community detection problems for networks with additional information," *JIFS*, vol. 39, no. 5, pp. 6217–6230, 2020.

[16] ——, "Multiple bipolar fuzzy measures: an application to community detection problems for networks with additional information," *IJCIS*, vol. 13, no. 1, pp. 1636–1649, 2020.

[17] D. Bader, A. Kappes, H. Meyerhenke, P. Sanders, C. Schulz, and D. Wagner.

[18] T. O. Kvalseth, "On normalized mutual information: measure derivations and properties," *Entropy*, vol. 19, no. 12, p. 631, 2017.

Explicaciones factuales y contrafactuales en árboles de clasificación difusos

Guillermo Tomás Fernández, Juan A. Aledo, Jose A. Gámez y Jose M. Puerta
Laboratorio de Sistemas Inteligentes y Minería de Datos (SIMD), I3A
Universidad de Castilla-La Mancha. Albacete, España
{Guillermo.Fernandez, JuanAngel.Aledo, Jose.Gamez, Jose.Puerta}@uclm.es

***Resumen*—Los algoritmos de clasificación actuales han alcanzado una gran popularidad debido a su eficiencia para generar modelos capaces de resolver problemas de alta complejidad. En particular, son los algoritmos denominados de caja negra los que mejores resultados ofrecen, ya que se benefician de esta enorme cantidad de datos para aprender modelos cada vez más precisos. Sin embargo, su principal desventaja frente a otros algoritmos más simples, p.e. un árbol de decisión, es la pérdida de interpretación tanto del modelo como de las clasificaciones individuales, lo que supone un grave inconveniente de cara a muchas aplicaciones en las que proveer una explicación es hoy día recomendable, e incluso obligatorio. Una práctica habitual es construir un modelo *explicable* que mimetice el comportamiento del clasificador más complejo en la zona circundante a la instancia a explicar.**

Sin embargo, la generación de explicaciones en estos modelos de caja blanca tampoco es trivial, lo que ha generado una intensa investigación en torno a ellos. Es habitual generar dos tipos de explicaciones, factuales y contrafactuales, que se complementan para informar al decisor por qué se ha clasificado la instancia en una determinada clase o categoría. En este trabajo proponemos la definición de explicaciones factuales y contrafactuales en el marco de los árboles de clasificación difusos, en los que al contrario de su contraparte *crisp* una instancia puede disparar más de una rama. Nuestra propuesta se centra en definir explicaciones factuales que contienen más de una regla, en contraposición al estándar habitual que se limita a incluir una única regla en la explicación factual. Además, introducimos la idea de explicación factual *robusta* y la generación de explicaciones contrafactuales a partir de la clasificación realizada y la explicación factual generada, que puede tener más de una regla.

***Index Terms*—Inteligencia artificial explicable (XAI); Lógica difusa; Árboles de decisión difusos; Explicaciones factuales; Explicaciones contrafactuales; Robustez.**

I. Introducción

La gran capacidad de decisión de los algoritmos modernos de clasificación ha originado un enorme incremento en la variedad de campos en los que se están aplicando dichos algoritmos. Problemas que hasta recientemente necesitaban obligatoriamente la intervención de un experto (o un sistema experto, con la complejidad y especificidad que este requiere) se están resolviendo mediante la aplicación de técnicas que se aprovechan de la gran cantidad de datos que se pueden recoger para aprender de manera automática un clasificador capaz de resolver el problema.

Sin embargo, a pesar de que estos algoritmos están alcanzando cotas de precisión cada vez más altas, lo hacen a costa de la interpretabilidad final para el usuario. El razonamiento que realizan estos sistemas es cada vez más complejo, lo que crea una necesidad de tener una creencia ciega en estos sistemas. En ciertos ámbitos críticos, esta creencia ciega no es suficiente para motivar el uso de este tipo de clasificadores. Este hecho se ve también refrendado por la legislación, p.e. el *derecho a la explicación*, incluido en la Regulación General de Protección de Datos aprobada por la Unión Europea [1], que no solo afecta a humanos sino también a técnicas de inteligencia artificial y sistemas informáticos.

Para atender las necesidades comentadas, surje la Inteligencia Artifical Explicable (XAI), una línea de investigación en auge que se centra en explicar aquellos modelos y sistemas que por sus características no resultan interpretables por un usuario. Dado que normalmente este tipo de algoritmos, denominados de *caja negra*, son los que mejores resultados obtienen en los distintos problemas, es especialmente interesante desarrollar sistemas capaces de explicar sus decisiones. En este sentido, los métodos *agnósticos* [2]–[4] explican la decisión realizada por un modelo complejo (habitualmente de caja negra) mediante la construcción de un modelo más sencillo de explicar que mimetiza al modelo complejo en la vecindad de la instancia cuya clasificación debe ser explicada.

Sin embargo, incluso cuando se construyen modelos de caja blanca, la generación de la explicación no es trivial, si no que es habitual generar explicaciones de distinto tipo: que indican por qué se ha clasificado en una determinada categoría (explicaciones factuales) y que indican por qué no ha sido clasificada en otra categoría (explicaciones contrafactuales) [5]. Existen distintas definiciones para este tipo de explicaciones en función del formalismo utilizado: árboles de decisión [4], [6], regresión [2], clasificadores probabilísticos [7], [8], etc.

En este trabajo nos centramos en el uso de clasificadores basados en árboles de clasificación difusos [9], en los que al contrario de sus homónimos *crisp*, una instancia puede disparar más de una regla. Parece razonable que si p.e. dos reglas difusas (extraídas del árbol) se disparan con similar grado de importancia (activación), ambas puedan ser usadas para explicar la clasificación, en lugar de seleccionar únicamente una de ellas. Partiendo de esta premisa nuestra contribución en este artículo se centra en la definición de explicaciones factuales que puedan contener, en caso de necesidad, más de una regla difusa. A partir de estas definiciones planteamos el concepto de *robustez* de una explicación factual y la generación de explicaciones contrafactuales a partir de la instancia a clasificar y la explicación factual (posiblemente no unitaria)

generada.

II. Definición del problema

Consideremos un problema de clasificación supervisada, que consiste en asignar una clase c_i perteneciente a un conjunto predefinido $C = \{c_1, \ldots, c_m\}$ a una instancia x. Sea esta instancia $x = (x_1, \ldots, x_n)$ una configuración de valores sobre n *variables predictoras*, $X_1, \ldots, X_n$.

Asumamos que asociada a cada variable predictora X_i hay una variable difusa (lingüística) $F_i = \{v_{i,1} \ldots, v_{i,k_i}\}$ definida mediante una partición de Ruspini de k_i conjuntos difusos ordenados. Utilizaremos $v_{i,j}$ para referirnos indistintamente tanto al conjunto difuso como a la etiqueta lingüística asociada al mismo. Dado un valor $\delta \in dom(X_i)$, sea

$$\mu_i(\delta) = (\mu_{i,1}(\delta), \ldots, \mu_{i,k_i}(\delta))$$

el vector de grados de pertenencia de δ a los k_i conjuntos difusos de F_i. Nótese que $\sum_{j=1}^{k_i} \mu_{i,j}(\delta) = 1$ (Figura 1). Definimos

$$f(\delta) = \arg \max_{1,\ldots,k_i} \mu_i(\delta),$$

como el índice correspondiente al conjunto difuso al que δ tiene mayor grado de pertenencia. En concreto, $\mu_{i,z_i}(\delta)$ es el grado de pertenencia del valor δ en F_i para el conjunto v_{i,z_i}.

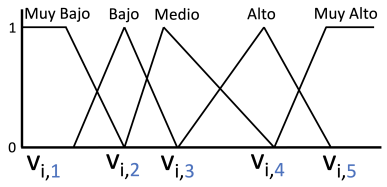


Figura 1. Ejemplo de partición de Ruspini con 5 conjuntos difusos para la variable i-ésima. Se muestran los conjuntos difusos $v_{i,j}$ y las etiquetas lingüísticas.

Por último, sea T un árbol de decisión difuso aprendido a partir de un conjunto de datos de entrenamiento $TR = \{(x_1^j, \ldots, x_n^j, c^j)\}_{j=1}^N$, donde N es el número de instancias y $c^j \in C$ es la clasificación (categoría o clase) asociada a cada instancia.

Nuestro objetivo es obtener una explicación $e = \langle R_f, R_{cf} \rangle$ para la instancia x a partir de las reglas[1] que se extraen del árbol T, donde R_f es una explicación factual y R_{cf} es una explicación contrafactual.

III. Estado del arte

III-A. *Explicaciones factuales y contrafactuales para clasificadores difusos*

En [6] se propone un método para generar explicaciones factuales y contrafactuales a partir de árboles de decisión tanto crisp como difusos. En ambos casos la explicación factual contiene una única regla y se genera una explicación contrafactural por cada categoría distinta a la predicha como clasificación.

[1]De forma más general se podría extender a un conjunto de reglas difusas.

A la hora de extraer los factuales, los autores consideran el árbol como una serie de nodos unidos por aristas pesadas. En un árbol crisp, la arista tendrá un peso binario, mientras que en un árbol difuso tendrá un valor real entre 0 y 1 (según el grado de activación de la instancia con el nodo). Estos nodos y aristas generan caminos desde la raíz hasta las hojas, cada uno con un peso asociado según el valor de las aristas que lo compongan. Finalmente, para obtener el factual los autores utilizan un α-corte en los caminos generados, de tal manera que todos los caminos con un peso menor que α se eliminen. Así, garantizan que sólo existe un único factual en un árbol difuso, de igual manera que en el caso crisp existe un único camino desde la raíz hasta la hoja que clasifica un ejemplo.

En cuanto a las explicaciones contrafactuales, existen $m-1$ explicaciones donde m es el número de clases. En particular, una explicación contrafactual es un camino desde la raíz del árbol hasta una hoja que tenga una clase distinta a la de la instancia a explicar.

III-B. *Árboles de Decisión Difusos*

En [9] se describe un método que primero genera de manera automática los conjuntos difusos que se utilizarán y posteriormente el proceso de aprendizaje del árbol usando dichos conjuntos. En este trabajo consideramos los árboles producidos por este algoritmo como clasificadores cuya inferencia se pretende explicar.

Dado un árbol T ya entrenado, cada rama tiene la forma

$$b : l_{i_1}[b] \wedge l_{i_2}[b] \wedge \cdots \wedge l_{i_b}[b] \wedge h[b]$$

donde

- i_b es el número de literales en el antecedente de la regla que se deriva de b.
- $l_i[b] : (F_i, v_{i,z_i})$ es un literal compuesto por la variable difusa F_i y la etiqueta v_{i,z_i}, con $1 \leq z_i \leq k_i$.
- $h[b] : \{c_1 = w_1[b], c_2 = w_2[b], \ldots, c_m = w_m[b]\}$ es un nodo hoja, donde se registra la importancia $w_i[b]$ para cada clase en esta hoja, con $\sum_{i=1}^m w_i[b] = 1$.

De cada rama b y cada clase c_i con $w_i[b] > 0$ obtenemos una regla

$$r_i[b] : l_{i_1}[b] \wedge l_{i_2}[b] \wedge \cdots \wedge l_{i_b}[b] \rightarrow c_i,$$

con peso asociado $w(r_i[b]) = w_i[b]$.

Denotaremos como $R(c_i)$ el conjunto de todas las reglas que se pueden extraer del árbol con consecuente c_i.

Llamamos **regla de peso máximo** para la rama b a

$$r[b] = \arg \max_{r_i[b]} w_i[b].$$

En caso de empate se elige una única regla de manera arbitraria o aleatoria.

Al realizar la inferencia, una instancia x dispara $p(x) \geq 1$ reglas (posiblemente de distintas ramas). Denotemos por $R(x) = \{r_1(x), \ldots, r_{p(x)}(x)\}$ el conjunto de reglas disparadas.

Se define el grado de emparejamiento de una regla $r(x) \in R(x)$ compuesta por i_b literales $l_i : (F_i, v_{i,z_i})$ con la instancia $x = (x_1, \ldots, x_n)$ como

$$md(r(x)) = \min_{i \in \{i_1,\ldots,i_b\}} (\mu_{i,z_i}(x_i))$$

Por otra parte, se define el grado de activación $AD(r(x))$ de una instancia x para la regla $r(x) \in R(x)$ como

$$AD(r(x)) = md(r(x)) \cdot w(r(x))$$

Dado un conjunto de reglas $\Gamma(x) \subseteq R(x)$, obtenemos la clase asociada a $\Gamma(x)$ mediante el método *weighted vote* como

$$c(\Gamma(x)) = \arg\max_{c_i} \sum_{\substack{r(x) \in \Gamma(x) \\ c(r(x)) = c_i}} AD(r(x)), \qquad (1)$$

con $c(r(x))$ la clase asociada a la regla $r(x)$. En caso de empate, este se rompe aleatoria o arbitrariamente. En particular, la clase $c(x)$ predicha por el conjunto de reglas $R(x)$ se obtiene como $c(x) = c(R(x))$.

IV. Metodología

En este trabajo, proponemos una definición de explicación factual que expande las encontradas hasta ahora en la literatura. Si bien las explicaciones factuales contienen una única regla y esto es razonable en entornos *crisp*, en el caso difuso es posible disparar más de una regla con grados de activación *similares*, por lo que elegir únicamente una de ellas puede suponer una pérdida de información imporante en la explicación facilitada. Proponemos una serie de definiciones que contemplan la posible inclusión de más de una regla en la explicación factual.

IV-A. Explicaciones factuales

Las reglas factuales son aquellas que nos ayudan a explicar la clasificación de una instancia en una determinada categoría. Dada una instancia x, y $c(x)$ su clasificación, sea $R'(c(x))$ el subconjunto de reglas de peso máximo en $R(x) \cap R(c(x))$, es decir, las reglas de peso máximo activadas por x con consecuente $c(x)$. Entonces, se define un factual $R_f(x)$ de x como un subconjunto de $R'(c(x))$ que explica la clasificación para la instancia x. Como caso particular, cuando $R'(c(x))$ sólo tiene una regla, esta será el factual.

IV-A1. Definiciones de factual: Sea x una instancia, y $|R'(c(x))| = p'(x)$. Sean $r_1(x), r_2(x), \ldots, r_{p'(x)}(x)$ los elementos de $R'(c(x))$ que suponemos ordenados de mayor a menor grado de activación, es decir, $AD(r_1(x)) \geq AD(r_2(x)) \geq \cdots \geq AD(r_{p'(x)}(x))$. A continuación proponemos tres formas de obtener un factual de x:

- Definimos el **factual asociado a la media**, $m\text{-}fact(x)$, como el subconjunto $R_f(x) = \{r_1(x), \ldots, r_q(x)\}$ formado por las q reglas de $R'(c(x))$ para las que se cumple

$$AD(r_j(x)) \geq \frac{\sum_{j=1}^{p'(x)} AD(r_j(x))}{p'(x)}, \forall j \in \{1, \ldots, q\} \quad (2)$$

- Dado $\lambda \in (0, 1]$, definimos el **factual asociado al λ-cociente**, $c(\lambda)\text{-}fact(x)$, como el subconjunto $R_f(x) = \{r_1(x), \ldots, r_q(x)\}$ formado por las primeras q reglas de $R'(c(x))$ para las que se cumple

$$\frac{AD(r_q(x))}{AD(r_{q+1}(x))} > 1 + \lambda \ \wedge \ \frac{AD(r_j(x))}{AD(r_{j+1}(x))} \leq 1 + \lambda \quad \forall j \in \{1, \ldots, q-1\} \quad (3)$$

- Dados $\lambda \in (0, 1]$ y $\beta \in (0, 1]$, definimos el **λ-cociente de masa mínima** β, $c(\lambda, \beta)\text{-}fact(x)$, como una variante de $c(\lambda)\text{-}fact(x)$ en la que exigimos que el sumatorio de todos los grados de activación de las reglas en el factual sea mayor que un umbral β. Es decir, el subconjunto $R_f(x) = \{r_1(x), \ldots, r_q(x)\}$ formado por las primeras q reglas de $R'(c(x))$, para las que se cumple:

$$\sum_{j=1}^{q} AD(r_j(x)) \geq \beta \quad \wedge \quad \frac{AD(r_q(x))}{AD(r_{q+1}(x))} > 1 + \lambda \quad (4)$$

IV-B. Robustez de una explicación factual

Sea

$$R_x(c_i) = \{r \in R(x) : c(r) = c_i\} = R(x) \cap R(c_i),$$

el conjunto de reglas activadas por la instancia x con consecuente c_i. Definimos

$$R_f^*(x) = R_f(x) \cup \bigcup_{\substack{i = 1, \ldots, m \\ c_i \neq c(x)}} R_x(c_i)$$

como el conjunto de reglas del factual y las disparadas por la instancia para el resto de las clases (independientemente de que sean o no de peso máximo).

Consideramos que un factual $R_f(x)$ es robusto cuando:

$$c(R_f^*(x)) = c(R_f(x))$$

En la literatura, la mayoría de métodos consideran la explicación factual para $c(x)$ como una única regla. Sin embargo, existen casos en los que esto podría no resultar robusto si el proceso de inferencia utiliza varias reglas para determinar la clasificación de la instancia. Este es otro motivo para la definición de explicaciones factuales que admitan múltiples reglas, aumentando así la robustez de la explicación.

IV-C. Explicaciones contrafactuales

Una explicación contrafactual para una instancia x es aquella que nos muestra los cambios mínimos que habría que realizar a x para que cambie de clase. Se generará una explicación contrafactual por cada clase alternativa. Proporcionamos distintos métodos.

1. Contrafactual con respecto a la instancia x y una clase $\overline{c} \neq c(x)$, $R_{cf}(x, \overline{c})$. Dada una clase $\overline{c} \in C$, denotaremos por $R'(\overline{c})$ el conjunto de todas las reglas del árbol de peso máximo y consecuente $\overline{c}$.

Definimos la distancia de r a la instancia x como

$$d(r, x) = \sum_{F_i \in V(r)} \left[(1 - \mu_{i,z_i}(x_i)) \ \cdot \ \frac{|z_i - f(x_i))|}{k_i} \right] \quad (5)$$

donde $V(r)$ es el conjunto de variables difusas que aparecen en la regla r y v_{i,z_i} el conjunto difuso asociado al literal de la variable F_i en r.

El factor $(1 - \mu_{i,z_i}(x_i))$ considera el grado de pertenencia de la instancia a cada uno de esos conjuntos difusos, de tal manera que no solo se tiene en cuenta la distancia al conjunto sino también si x pertenece al mismo.

En cuanto al cociente $\frac{|z_i - f(x_i))|}{k_i}$, tiene en cuenta la distancia entre el índice del conjunto v_{i,z_i} del literal y el índice del conjunto al que la instancia x tiene mayor grado de pertenencia para la variable F_i.

Definimos la explicación contrafactual $R_{cf}(x, \overline{c})$ como

$$\arg \min_{r \in R'(\overline{c})} d(r, x)$$

2. Contrafactual con respecto al factual $R_f(x)$ y una clase $\overline{c} \neq c(x)$, $R_{cf}(x, R_f(x), \overline{c})$. Nuestro objetivo ahora es generar una explicación contrafactual que se diferencie lo mínimo posible de la explicación factual, pero clasifique la instancia x en $\overline{c}$.

Comencemos definiendo la distancia entre una regla contrafactual r y una regla ρ del factual $R_f(x)$ con $V(r) = V(\rho)$ como

$$d_{rule}(r, \rho, x) = \sum_{F_i \in V(r)} \left[(1 - \mu_{i,z_{r_i}}(x_i)) \cdot \frac{|z_{r_i} - z_{\rho_i}|}{k_i} \right] \quad (6)$$

con $v_{i,z_{r_i}}$ (resp. $v_{i,z_{\rho_i}}$) el conjunto difuso asociado al literal de la variable F_i en r (resp. ρ).

Por otra parte, dado $S \subseteq V(r)$, denotamos por $r^{\downarrow S}$ la simplificación de la regla r resultante de eliminar en el antecedente los literales asociados a las variables que no aparecen en el subconjunto S.

Definimos entonces la distancia entre r y $R_f(x) = \{\rho_1, \ldots, \rho_q\}$ como

$$d(r, R_f(x)) = \sum_{\rho_j \in R_f(x)} [(1 - md(\rho_j(x))) \cdot cf\_dist(r, \rho_j, x)] \quad (7)$$

con

$$\begin{aligned} cf\_dist(r, \rho_j, x) =& |V(r) \cup V(\rho_j)| - |V(r) \cap V(\rho_j)| + \\ & d_{rule}(r^{\downarrow V(r) \cap V(\rho_j)}, r_i^{\downarrow V(r) \cap V(\rho_j)}, x) \end{aligned} \quad (8)$$

donde $|V(r) \cup V(\rho_j)| - |V(r) \cap V(\rho_j)|$ es el número de variables difusas diferentes entre las reglas r y ρ_j

Es decir, $d(r, R_f(x))$ es la suma de la distancia contrafactual cf_dist (Ecuación 8) de r a cada $\rho_j \in R_f(x)$, ponderadas por el grado de emparejamiento md de estas reglas ρ_j a la instancia x. De este modo tendrá más influencia una regla contrafactual que esté muy cerca de la "mejor" regla del factual que una que esté muy cerca de la "peor").

Al calcular este tipo de distancias, se puede interpretar de manera distinta qué significa "realizar el mínimo número de cambios". Hay dos tipos de cambios diferenciados entre dos reglas: una modificación de una variable que ya existe; y una adición o substracción de una variable que no existe. En este trabajo, se le ha dado más importancia a añadir o eliminar variables, haciendo que su contribución a la distancia sea mayor. El razonamiento detrás de esta decisión es que dada una regla, la modificación de una variable que ya existe en la regla es una menor perturbación que considerar una variable nueva, haciendo así que la longitud de la regla se modifique.

Finalmente, definimos la explicación contrafactual $R_{cf}(x, R_f(x), \overline{c})$ como

$$\arg \min_{r \in R'(\overline{c})} d(r, R_f(x)).$$

V. Ejemplo ilustrativo

V-A. *Instancia a través del árbol difuso*

Para ilustrar los conceptos introducidos se ha entrenado un árbol de decisión difuso usando el método propuesto en [9] para el conjunto de datos wine [10], definido mediante 13 atributos numéricos y una clase con tres categorías ($\{type\ 0, type\ 1, type\ 2\}$. Para el ejemplo hemos seleccionado una instancia que mostramos a continuación indicando el grado de pertenencia a las variables difusas aprendidas para la construcción del árbol (se le han asignado etiquetas del conjunto (Muy Bajo, Bajo, Medio, Alto, Muy Alto)). Por claridad, solo se muestran las variables que aparecen en las reglas disparadas y explicaciones generadas. Se usa el formato: F_i : {"$v_{i,1}$": $\mu_{v_{i,1}}(x_i), \cdots$, "$v_{i,k_i}$": $\mu_{v_{i,k_i}}(x_i)$} para los k_i conjuntos difusos de cada variable, y se muestran sólo las variables que aparecen en las reglas por claridad:

$$\begin{aligned} flavanoids &: \{\text{“}muy\ bajo\text{”} : 0.14, \text{“}bajo\text{”} : 0.86, \\ &\quad \text{“}alto\text{”} : 0, \text{“}muy\ alto\text{”} : 0\}, \\ color\_intensity &: \{\text{“}bajo\text{”} : 0.357, \text{“}medio\text{”} : 0.643, \\ &\quad \text{“}alto\text{”} : 0\}, \\ alcohol &: \{\text{“}muy\ bajo\text{”} : 0, \text{“}bajo\text{”} : 0.616, \\ &\quad \text{“}alto\text{”} : 0.384, \text{“}muy\ alto\text{”} : 0\}, \\ hue &: \{\text{“}bajo\text{”} : 0.45, \text{“}medio\text{”} : 0.55, \\ &\quad \text{“}alto\text{”} : 0\}, \\ proanthocyanins &: \{\text{“}bajo\text{”} : 0.64, \text{“}medio\text{”} : 0.36, \\ &\quad \text{“}alto\text{”} : 0\} \\ od280/od315 &: \{\text{“}bajo\text{”} : 0.274, \text{“}medio\text{”} : 0.726, \\ &\quad \text{“}alto\text{”} : 0\} \\ class &: type\ 2 \end{aligned}$$

V-B. *Factual*

Utilizando la Ecuación 2 para obtener el factual $R_f(x)$, resultan cuatro reglas (consecuente = $type\ 2$):

$$\begin{aligned} r_1: &\ (flavanoids\ bajo) \wedge (color\_intensity\ medio) \wedge \\ &(alcohol\ bajo) \wedge (hue\ bajo), w(r_1) = 1 \\ &AD(r_1(x)) = 0.45 \\ r_2: &\ (flavanoids\ bajo) \wedge (color\_intensity\ medio) \wedge \\ &(alcohol\ alto) \wedge (proanthocyanins\ bajo), w(r_2) = 1 \\ &AD(r_2(x)) = 0.38 \\ r_3: &\ (flavanoids\ bajo) \wedge (color\_intensity\ bajo) \wedge \\ &(alcohol\ alto) \wedge (od280/od315\ medio), w(r_3) = 1 \\ &AD(r_3(x)) = 0.36 \\ r_4: &\ (flavanoids\ bajo) \wedge (color\_intensity\ medio) \wedge \\ &(alcohol\ alto) \wedge (proanthocyanins\ medio), w(r_4) = 0.92 \\ &AD(r_4(x)) = 0.33 \end{aligned}$$

Se muestra además el grado de activación (Sección III-B). A modo de ejemplo, para r_4 sería

$$\begin{aligned} AD(r_4(x)) &= md(r_4(x)) \cdot w(r_4(x)) \\ &= min(0.86, 0.643, 0.384, 0.36) \cdot 0.92 = 0.33 \end{aligned}$$

Podemos ver que las cuatro reglas son ciertamente parecidas y con un grado de activación similar. En función de una ligera variación de $alcohol$ y $color\_intensity$, la variable que explicaría la clasificación de la instancia cambia (hue, $proanthocyanins$ u $od280/od315$).

Probablemente para el usuario sea más interesante tener esta información más completa que únicamente la regla r_1.

V-C. Robustez

Para comprobar la robustez del factual, el primer paso es comprobar si existen reglas del árbol que se disparan para el resto de clases ($type$ 0 y/o $type$ 1). En este caso, se disparan suficientes reglas de la clase $type$ 1 para alcanzar un grado de activación acumulado igual a 1.22. Este hecho provocaría que si únicamente incluyéramos r_1 como explicación factual, dicho factual no sería robusto. Sin embargo, el factual formado por $\{r_1, r_2, r_3, r_4\}$ sí lo es puesto que $\sum_{i=1}^{4} AD(r_i(x)) = 1.52$.

V-D. Contrafactual

Dado que el clasificador obtiene el valor $type$ 2 para la clase, pueden existir explicaciones contrafactuales para las clases $type$ 0 y $type$ 1. Usaremos la segunda definición propuesta en la Sección IV-C.

V-D1. Clase type 1: Consideramos todas las reglas de peso máximo con consecuente $type$ 1. Las dos más cercanas al factual en función de la distancia definida son:

$$\begin{aligned} r_{c1}: &(flavanoids\ bajo) \wedge (color\_intensity\ medio) \wedge \\ &(alcohol\ alto) \wedge (proanthocyanins\ alto) \\ &d(r_{c1}, R_f(x)) = 12.98 \\ r_{c2}: &(flavanoids\ bajo) \wedge (color\_intensity\ medio) \wedge \\ &(alcohol\ muy\ alto) \wedge (proanthocyanins\ alto) \\ &d(r_{c2}, R_f(x)) = 14.68 \end{aligned}$$

De estas dos reglas, sería r_{c1} el contrafactual para la clase $type$ 1.

V-D2. Clase type 0: Razonando igual, las dos reglas que minimizan la distancia a la explicación factual son

$$\begin{aligned} r_{c1}: &(flavanoids\ alto) \wedge (alcohol\ muy\ alto) \\ &d(r_{c1}, R_f(x)) = 24.18 \\ r_{c2}: &(flavanoids\ muy\ alto) \wedge (alcohol\ alto) \\ &d(r_{c2}, R_f(x)) = 24.72 \end{aligned}$$

Igual que antes, r_{c1} será el contrafactual para la clase $type$ 0.

VI. Evaluación

VI-A. Conjuntos de datos

Los experimentos se han realizado con los conjuntos de datos *iris* [10], [11], *wine* [10], [12] y *beer* [13]. Los dos primeros conjuntos de datos se utilizan comúnmente en problemas de clasificación y el tercero se ha usado en XAI [**?**].

VI-B. Metodología experimental

Para cada conjunto de datos se ha entrenado un árbol de decisión difuso usando *holdout* como metodología de validación (70 % entrenamiento y 30 % test). Así, las instancias cuya clasificación debe ser explicada son las del conjunto de *test*.

VI-C. Recursos computacionales

Todos los algoritmos han sido programados en Python 3.8, debido a la potencia de librerías como *scikit-learn* [14] que ayudan con el tratamiento de datos y las estructuras necesarias. Para el algoritmo de aprendizaje del árbol de clasifiación y la definición de las variables difusas se ha seguido [9], en particular, la implementación del árbol FMDT (*Fuzzy Multiway Decision Tree*). Posteriormente, se han añadido los métodos necesarios para calcular las explicaciones factuales y contrafactuales propuestas en este artículo. Para garantizar la reproducibilidad de las pruebas realizadas, tanto el código como (el acceso a) los datos se publicará en un repositorio abierto en Github.

VI-D. Explicaciones factuales y robustez.

En la Tabla I se muestra información sobre las explicaciones factuales obtenidas y su robustez. En el caso de los criterios que dependen de umbrales (λ, β) se han probado varios valores. Las columnas de la tabla representan:

- **Config**: Método para obtener la explicación factual (Sección IV-A1) y parámetros utilizados.
- $q > 1$: Proporción de instancias (del conjunto de test) para las que la explicación factual incluye más de una regla.
- **NR-Fact**: Proporción de instancias (del conjunto de test) cuya explicación no es robusta.
- **Len**: Longitud media de las explicaciones factuales (medida en número de reglas).

Además, se calculan las siguientes constantes, que son independientes del método considerado para generar la explicación factual:

- **ExistCF**: Proporción de instancias (del conjunto de test) para las que $R(x)$ contiene reglas con consecuente distinto a la clase predicha ($c(x)$).
- **NR-Rule**: Proporción de instancias (del conjunto de test) para las que la explicación factual únicamente por la regla con mayor grado de activación no es robusto.

Tabla I
EXPERIMENTOS SOBRE LA ROBUSTEZ

Configuración	Iris $q>1$	ExistCF: 0.6 NR-Fact	NR-Rule: 0.13 Len	Wine $q>1$	ExistCF: 0.677 NR-Fact	NR-Rule: 0.1 Len	Beer $q>1$	ExistCF: 0.625 NR-Fact	NR-Rule: 0.1 Len
$m\text{-}fact(x)$	0.2	0.06	1.22	0.559	0.025	2.05	0.475	0.1	1.58
$c(\lambda)\text{-}fact(x)$ $\lambda: 0.1$	0.06	0.06	1.1	0.0508	0.075	1.033	0.175	0.1	1.2
$c(\lambda)\text{-}fact(x)$ $\lambda: 0.25$	0.12	0.06	1.22	0.237	0.025	1.644	0.225	0.08	1.325
$c(\lambda,\beta)\text{-}fact(x)$ $\lambda: 0.1$ $\beta: 0.5$	0.06	0.06	1.1	0.0508	0.075	1.033	0.188	0.1	1.213
$c(\lambda,\beta)\text{-}fact(x)$ $\lambda: 0.1$ $\beta: 0.7$	0.28	0.03	1.34	0.13	0	1.305	0.3	0.04	1.4
$c(\lambda,\beta)\text{-}fact(x)$ $\lambda: 0.25$ $\beta: 0.5$	0.12	0.06	1.22	0.237	0.025	1.644	0.225	0.085	1.325
$c(\lambda,\beta)\text{-}fact(x)$ $\lambda: 0.25$ $\beta: 0.7$	0.34	0.03	1.64	0.254	0	1.711	0.325	0.04	1.475

En la Tabla I se puede observar que la media suele ser el método que obtiene reglas más largas, sin que ello necesariamente implique que sean más robustas. Los otros dos métodos, que tienen en cuenta la similaridad entre las reglas del factual, obtienen unas reglas algo más cortas y además suelen ser más robustas. Además, en todas las bases de datos se ve que existen más reglas que no son robustas a factuales, independientemente de cómo estén construidos. Así, se demuestra una necesidad de buscar esta robustez con múltiples reglas en el factual.

VI-E. Explicaciones contrafactuales

Evaluaremos los siguientes parámetros:

- **Número de contrafactuales (NumCF)**. El número de posibles contrafactuales por clase. Un mayor número de posibles contrafactuales representa un mayor factor de ramificación del árbol.
- **Mejor distancia mínima (BestMinDist)**. La distancia del mejor contrafactual a la explicación factual.
- **Longitud del Contrafactual (CFLength)**. Calculada como el número medio de literales que contiene el contrafactual.

En la Tabla II se muestran estos datos para los distintos métodos propuestos para calcular la explicación contrafactual (**Configuración (Config)**). Se empleará una única configuración de parámetros.

Tabla II
EXPERIMENTOS SOBRE LOS CONTRAFACTUALES.

Configuración	Iris CFLen	Iris NumCF: 3.73 BestMinDist	Wine CFLen	Wine NumCF: 3.73 BestMinDist	Beer CFLen	Beer NumCF: 9.37 BestMinDist
CF de x	1.69	2.42	3.86	8.97	2.62	0.95
CF de $m\text{-}fact(x)$	1.53	0.391	2.61	1.93	2.203	0.721
CF de $c(\lambda)\text{-}fact(x)$ $\lambda : 0.1$	1.53	0.339	2.79	0.767	2.191	0.477
CF de $c(\lambda, \beta)\text{-}fact(x)$ $\lambda : 0.1$ $\beta : 0.7$	1.53	0.72	2.56	2.11	2.2	0.93

De los resultados podemos observar que en general los contrafactuales que se extraen con respecto a la instancia tienen una longitud ligeramente superior que lo generados con respecto a la explicación factual. Esto se debe a que al estar la instancia definida sobre todos los atributos, no penaliza necesariamente que la regla contrafactual tenga más literales, sino que tenga menor distancia a sus valores (conjuntos difusos). Por otro lado, la gran diferencia de distancia entre los contrafactuales $m\text{-}fact$ y los contrafactuales $c\text{-}fact$ se debe a que las explicaciones factuales $c\text{-}fact$ contienen más reglas. Al calcularse la distancia como la suma de la distancia a todas las reglas del factual, cuanto mayor sea el factual (como se observa en la Tabla I) mayor será la distancia.

VII. CONCLUSIONES Y TRABAJO FUTURO

Partiendo de la hipótesis de que en un entorno de razonamiento difuso generar explicaciones factuales con una única regla puede suponer una pérdidad de información, se han propuesto definiciones alternativas que permiten incluir múltiples reglas. Como se ha comprobado en los experimentos, en la mayoría de los casos esto no es necesario, pero sí en un porcentaje nada despreciable. Se ha introducido además el concepto de robustez para la explicación factual, comprobándose en la evaluación realizada la mejoría que en este sentido supone introducir más de una regla (en los casos más inciertos). Finalmente se han propuesto definiciones de explicación contrafactual ligadas a las proporcionadas para las explicaciones factuales.

Por tratarse de un primer trabajo en esta línea de investigación, es obvio que existen distintas vías de ampliación, como es su experimentación/integración en métodos agnósticos de explicación. Concretamente, pretendemos generalizar el estudio presentado en [4] al caso de los árboles de clasifiación difusos. Consideramos además de interés estudiar métodos de simplificación de las explicaciones factuales generadas, de forma que estas sean lo más compactas posible, mejorando así su comprensión por parte de los usuarios.

AGRADECIMIENTOS

Este trabajo ha sido parcialmente financiado por el Gobierno de España, fondos FEDER y la Junta de Comunidades de Castilla-La Mancha mediante los proyectos PID2019–106758GB–C33 /AEI/ 10.13039/501100011033 y SBPLY/17/180501/000493. Guillermo Tomás Fernández también está financiado por FPU19/02930 del MCIU.

REFERENCIAS

[1] G. D. P. Regulation, "General data protection regulation (gdpr)," Intersoft Consulting, Accessed in October 24, vol. 1, 2018.

[2] M. T. Ribeiro, S. Singh, and C. Guestrin, ""why should i trust you?" explaining the predictions of any classifier," in Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, 2016, pp. 1135–1144.

[3] ——, "Anchors: High-precision model-agnostic explanations," in Thirty-Second AAAI Conference on Artificial Intelligence, 2018.

[4] R. Guidotti, A. Monreale, F. Giannotti, D. Pedreschi, S. Ruggieri, and F. Turini, "Factual and counterfactual explanations for black box decision making," IEEE Intell. Syst., vol. 34, no. 6, pp. 14–23, 2019.

[5] I. Stepin, J. M. Alonso, A. Catalá, and M. Pereira-Fariña, "A survey of contrastive and counterfactual explanation generation methods for explainable artificial intelligence," IEEE Access, vol. 9, pp. 11 974–12 001, 2021.

[6] I. Stepin, J. M. Alonso, A. Gatala, and M. Pereira-Fariña, "Generation and evaluation of factual and counterfactual explanations for decision trees and fuzzy rule-based classifiers," in 2020 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE). IEEE, 2020, pp. 1–8.

[7] A. Shih, A. Choi, and A. Darwiche, "A symbolic approach to explaining bayesian network classifiers," in Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden, J. Lang, Ed., 2018, pp. 5103–5111.

[8] E. Albini, A. Rago, P. Baroni, and F. Toni, "Relation-based counterfactual explanations for bayesian network classifiers," in Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020, C. Bessiere, Ed., 2020, pp. 451–457.

[9] A. Segatori, F. Marcelloni, and W. Pedrycz, "On distributed fuzzy decision trees for big data," IEEE Transactions on Fuzzy Systems, vol. 26, no. 1, pp. 174–192, 2017.

[10] D. Dua and C. Graff, "UCI machine learning repository," 2017. [Online]. Available: http://archive.ics.uci.edu/ml

[11] R. A. Fisher, "The use of multiple measurements in taxonomic problems," Annals of eugenics, vol. 7, no. 2, pp. 179–188, 1936.

[12] M. Forina et al., "An extendible package for data exploration, classification and correlation. institute of pharmaceutical and food analysis and technologies," 1998.

[13] G. Castellano, C. Castiello, and A. M. Fanelli, "The fisdet software: Application to beer style classification," in 2017 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE). IEEE, 2017, pp. 1–6.

[14] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," Journal of Machine Learning Research, vol. 12, pp. 2825–2830, 2011.

Comparando variabilidades de conjuntos aleatorios

Juan Jesús Salamanca
Departamento de Estadística e I.O. y D.M.
Universidad de Oviedo
Gijón, España
salamancajuan@uniovi.es

Susana Montes
Departamento de Estadística e I.O. y D.M.
Universidad de Oviedo
Oviedo, España
montes@uniovi.es

***Resumen*—Los conjuntos aleatorios son modelos de probabilidad destacados en Probabilidades Imprecisas. Un problema que nos ocupa es el de establecer alguna decisión en un ambiente de incertidumbre. Aquí, los ordenes estocásticos juegan un papel fundamental, ya que suponen técnicas de decisión en este contexto. En esta contribución introducimos un nuevo orden dispersivo para conjuntos aleatorios. Es decir, un método de preferencia entre dos conjuntos aleatorios que está basado en la variabilidad de los elementos aleatorios. Además, estudiaremos brevemente las propiedades más destacadas que presenta este orden.**

***Index Terms*—conjunto aleatorio, orden dispersivo, decisión bajo imprecisión**

I. Introducción

Los órdenes estocásticos son herramientas muy útiles en Estadística (véanse las referencias [4] y [6], por ejemplo). Dentro de estos órdenes nos encontramos los órdenes dispersivos, que son aquellas herramientas que permiten comparar dos elementos aleatorios en términos de sus variabilidades.

Así, encontramos en primer lugar al orden dispersivo usual.

Definición 1. *[6, Sec. 3.B] Dadas dos variables univariantes X e Y, el* orden dispersivo usual *elige X sobre Y si*

$$F_X^{-1}(\alpha) - F_X^{-1}(\beta) \leq F_Y^{-1}(\alpha) - F_Y^{-1}(\beta)$$

donde F_X^{-1} y F_Y^{-1} denotan las funciones inversas continuas por la derecha de las funciones de distribución F_X y F_Y de X e Y, respectivamente.

El principal escollo que presenta esta técnica es su dificultad para ser extendida a un punto aleatorio de un espacio métrico arbitrario.

Por otro lado encontramos el orden dispersivo débil.

Definición 2. *[2] Dados dos puntos aleatorios X e Y de un espacio métrico $(\mathcal{M}, d)$, diremos que X* es preferido a Y en el orden dispersivo débil *si se satisface*

$$\mathbb{P}\{d(X, X') \leq r\} \leq \mathbb{P}\{d(Y, Y') \leq r\}$$

para todo $r \geq 0$, donde X' e Y' son puntos aleatorios idénticamente distribuidos a X e Y, respectivamente y $\mathbb{P}$ denota la medida de probabilidad subyacente.

Una dificultad que presenta esta otra técnica es que no produce comparación en numerosas ocasiones.

Otros órdenes dispersivos se basan en contracciones. Sin embargo, tales aplicaciones pueden no tener sentido para algunas clases de espacios topológicos (véase [5]).

Finalmente nos encontramos con el orden dispersivo por bolas.

Definición 3. *[5] Dados dos puntos aleatorios X e Y de un espacio métrico $(\mathcal{M}, d)$, se dice que X es* preferido a Y en el orden dispersivo por bolas *si para todo punto p existe otro punto q tal que*

$$\mathbb{P}\{X \in \mathbb{B}_q(r)\} \geq \mathbb{P}\{Y \in \mathbb{B}_p(r)\}$$

se satisface para todo $r \geq 0$, donde $\mathbb{B}_x(r)$ denota la bola (cerrada) de centro x y radio r.

Podemos interpretar este orden diciendo que, para todo punto p, la distribución de $d(Y, p)$ está más dispersa que la distribución de $d(X, q)$ para algún punto q. Es decir, controlamos la variabilidad de los puntos aleatorios a través de bolas geométricas. Esta última técnica tiene la gran ventaja de ser lo suficientemente maleable como para poder extenderse a otros esquemas de trabajo.

Por otro lado, conviene notar que en numerosas ocasiones la información disponible es difusa, vaga o imprecisa en general. Este es el punto de partida de lo que se ha convenido en llamar teoría de Probabilidades Imprecisas (véase [7]). En líneas generales, esta rama de la Estadística estudia modelos de probabilidad bajo imprecisión e incertidumbre. En este esquema de trabajo, los conjuntos aleatorios, en su interpretación disyuntiva u óntica, son modelos destacados de probabilidades imprecisas (véase [1]).

En este contexto, necesitamos alguna herramienta que nos permita elegir como preferido un conjunto aleatorio frente a otro en términos de sus variabilidades. En esta contribución

Este trabajo ha sido parcialmente financiado por el proyecto PGC2018-098623-B-I00 del Ministerio de Ciencia, Innovación y Universidades. Los autores quieren agradecer dicha financiación.

presentaremos un orden dispersivo nuevo para conjuntos aleatorios. Este orden dispersivo está basado en una comparación de las funciones de plausibilidad de los conjuntos aleatorios, con lo que se puede entender como una extensión del orden dispersivo por bolas anteriormente comentado.

Recuérdese que la función de plausibilidad caracteriza al conjunto aleatorio, tal y como se recoge en el celebrado teorema de Choquet (véase [3] y las referencias allí citadas). En este sentido contemplamos la información estadística que poseen los conjuntos aleatorios a comparar.

II. Un nuevo orden dispersivo para conjuntos aleatorios

Recordamos en primer lugar la noción de conjunto aleatorio cerrado. Para ello, comencemos fijando un espacio topológico Hausdorff, segundo-contable y localmente compacto $\mathcal{M}$, al que le dotaremos de una distancia d. La distancia nos permite considerar las bolas (geométricas); así, denotaremos por $\mathbb{B}_p(r)$ a la bola de centro p y radio r. Al mismo tiempo denotemos a la clase de conjuntos cerrados de $\mathcal{M}$ por $\mathcal{F}$.

Definición 4. *[3] Un* conjunto aleatorio (cerrado) $\widehat{X}$ *sobre* $\mathcal{M}$ *es una aplicación* $\widehat{X} : (\Omega, \sigma, \mathbb{P}) \to \mathcal{F}$ *tal que*

$$\left\{w \in \Omega : \widehat{X}(w) \cap K \neq \emptyset\right\} \in \sigma$$

donde K *es un conjunto compacto de* $\mathcal{M}$ *y* $(\Omega, \sigma, \mathbb{P})$ *es un espacio de probabilidad completo.*

De la definición anterior se deduce inmediatamente que está bien definida la aplicación $Pl_{\widehat{X}}$ que asocia a cada compacto K de $\mathcal{M}$ la probabilidad $Pl_{\widehat{X}}(K) = \mathbb{P}\left\{w \in \Omega : \widehat{X}(w) \cap K \neq \emptyset\right\}$. Esta aplicación se llama *función de plausibilidad* (del conjunto aleatorio).

El celebrado resultado de Choquet establece que todo conjunto aleatorio queda caracterizado por su función de plausibilidad (véase [3]).

En este trabajo estamos interesados en obtener una técnica de comparación entre dos conjuntos aleatorios en términos dispersivos. Para tal fin, extenderemos en este sentido el orden dispersivo que se da en [5] para puntos aleatorios.

Definición 5. *Sean* $\widehat{X}$ *e* $\widehat{Y}$ *dos conjuntos aleatorios de* $\mathcal{M}$. *Diremos que* $\widehat{X}$ *es* preferido a $\widehat{Y}$ en el orden dispersivo de plausibilidad *si para todo punto* p *existe otro punto* q *tal que*

$$Pl_{\widehat{X}}\left(\mathbb{B}_q(r)\right) \geq Pl_{\widehat{Y}}\left(\mathbb{B}_p(r)\right) \tag{1}$$

se satisface para todo $r \geq 0$, *donde* $Pl_{\widehat{X}}$ *y* $Pl_{\widehat{Y}}$ *denotan las funciones de plausibilidad de* $\widehat{X}$ *e* $\widehat{Y}$, *respectivamente.*

Es claro observar que el orden introducido solo tiene en cuenta el grado de dispersión de los conjuntos aleatorios. De hecho, recoge la información acerca de cómo se aleja el conjunto aleatorio respecto de puntos. Si un conjunto aleatorio es preferido a otro, entonces podemos decir que, en algún sentido, se acerca más a un punto el primero que el segundo.

Las propiedades más destacables de esta nueva técnica se recogen a continuación. Antes de ello, recordemos el concepto de isometría, que generaliza a las traslaciones y rotaciones de un espacio euclídeo. Dado un espacio métrico $(\mathcal{M}, d)$, una *isometría* ϕ es una aplicación $\phi : \mathcal{M} \to \mathcal{M}$ biyectiva que conserva distancias, es decir, que para cualesquiera puntos p y q, se tiene: $d(p, q) = d(\phi(p), \phi(q))$. Dado un conjunto aleatorio $\widehat{X}$ definimos el conjunto aleatorio $\phi \circ \widehat{X}$ como: $\left(\phi \circ \widehat{X}\right)(w) = \phi\left(\widehat{X}(w)\right)$, para todo w del espacio de probabilidad subyacente.

Proposición 1. *El orden dispersivo de plausibilidad presenta las siguientes propiedades:*

(a) *Es transitivo. De forma más precisa, sean* $\widehat{X}$, $\widehat{Y}$ *y* $\widehat{Z}$ *tres conjuntos aleatorios de* $\mathcal{M}$. *Si* $\widehat{X}$ *es preferido a* $\widehat{Y}$ *e* $\widehat{Y}$ *es preferido a* $\widehat{Z}$, *entonces* $\widehat{X}$ *es preferido a* $\widehat{Z}$.

(b) *Es reflexivo y, por tanto, un preorden, pero no es un orden.*

(c) *Es compatible con isometrías. De forma más precisa, sea* ϕ *una isometría de* $(\mathcal{M}, d)$. *Entonces, el conjunto aleatorio* $\widehat{X}$ *es preferido al conjunto aleatorio* $\widehat{Y}$ *si y solo si* $\phi \circ \widehat{X}$ *es preferido a* $\widehat{Y}$ *si y solo si* $\widehat{X}$ *es preferido a* $\phi \circ \widehat{Y}$.

Idea de la demostración.- Para demostrar este resultado hay que tener en cuenta:

(a) La propiedad de transitividad se obtiene a partir de la transitividad que se hereda de la comparación de las correspondientes funciones de plausibilidad, y haciendo uso de las definiciones.

(b) Es trivial que es reflexivo, sin más que considerar como q el valor de p. Sin embargo se pueden encontrar dos conjuntos aleatorios distintos tales que cualquiera de los dos sea preferido al otro.

(c) Se puede deducir a partir del siguiente hecho:

$$\begin{aligned}
&Pl_{\phi\circ\widehat{X}}\left(\mathbb{B}_p(r)\right) = \mathbb{P}\left\{w : \phi \circ \widehat{X}(w) \cap \mathbb{B}_p(r) \neq \emptyset\right\} = \\
&\mathbb{P}\left\{w : \phi^{-1}\left(\phi \circ \widehat{X}(w) \cap \mathbb{B}_p(r)\right) \neq \emptyset\right\} = \\
&\mathbb{P}\left\{w : \widehat{X}(w) \cap \phi^{-1} \circ \mathbb{B}_p(r) \neq \emptyset\right\} = \\
&\mathbb{P}\left\{w : \widehat{X}(w) \cap \mathbb{B}_{\phi^{-1}(p)}(r) \neq \emptyset\right\} = \\
&Pl_{\phi\circ\widehat{X}}\left(\mathbb{B}_{\phi^{-1}(p)}(r)\right).
\end{aligned}$$

En este sentido, la isometría ϕ conserva la función de plausibilidad.

Por un lado, la propiedad de transitividad es una propiedad muy importante que un buen orden debe presentar. Por otro lado, que el orden sea compatible con las isometrías está asociado a su carácter de ser un orden dispersivo, puesto

que no tiene en cuenta "posiciones" del conjunto aleatorio, sino cómo puede alejarse o acercarse a un punto dado. De hecho, recuérdese que el concepto de isometría generaliza propiamente al grupo de traslaciones y rotaciones de un espacio euclídeo. En particular, esta propiedad indica que el orden es invariante frente a estas transformaciones de un espacio euclídeo.

El orden que estamos introduciendo ha de proveer argumentos lógicos y de peso para el/la decisor/a. A continuación daremos un argumento para tal fin.

Dado que estamos trabajando en términos de probabilidades imprecisas, el conjunto aleatorio se puede interpretar como un modelo que engloba un conjunto de modelos de probabilidad exacta. En concreto, una selección no es sino un modelo de probabilidad exacto que es compatible con la información dada por el conjunto aleatorio, tal como se deduce de la siguiente definición.

Definición 6. *[3] Una* selección X *de un conjunto aleatorio* $\widehat{X}$ *sobre* $(\mathcal{M}, d)$ *es un punto aleatorio de* $\mathcal{M}$ *tal que* $X(w) \in \widehat{X}(w)$ *casi seguro.*

Podemos dar el siguiente resultado, que se particulariza para conjuntos aleatorios discretos (en esta contribución, aquellos cuyo espacio de probabilidad es discreto, por definición).

Teorema 1. *Sean* $\widehat{X}$ *e* $\widehat{Y}$ *dos conjuntos aleatorios discretos sobre* $(\mathcal{M}, d)$*. Si* $\widehat{X}$ *es preferido a* $\widehat{Y}$ *en el orden dispersivo de plausibilidad, entonces para todo punto* p *y para toda selección* Y *de* $\widehat{Y}$ *podemos encontrar un punto* q *y una selección* X *de* $\widehat{X}$ *tal que*

$$\mathbb{E}[d^2(Y,p)] \geq \mathbb{E}[d^2(X,q)].$$

En particular, para toda selección de $\widehat{Y}$ *se puede encontrar una selección de* $\widehat{X}$ *con menor o igual varianza.*

Idea de la demostración.- Por reducción al absurdo. Asumamos que existe un punto p' y una selección Y' de $\widehat{Y}$ tales que $\mathbb{E}[d^2(Y',p')] \geq \mathbb{E}[d^2(X,q)]$ se satisface para todo punto q y selección X de $\widehat{X}$. Esto implica que existe un real no-negativo r para el cual $\mathbb{P}\{d(Y',p') \leq r\} > \mathbb{P}\{d(X,q)\}$ se satisface para todo punto q y selección X de $\widehat{X}$. Nótese que la función de plausibilidad satisface $Pl_{\widehat{Y}}(\mathbb{B}_{p'}(r)) \geq \mathbb{P}\{d(Y',p') \leq r\}$. El siguiente paso es usar que $\widehat{X}$ es preferido a $\widehat{Y}$ para deducir que debe existir un punto q' tal que $Pl_{\widehat{X}}(\mathbb{B}_{q'}(r)) > \mathbb{P}\{d(X,q') \leq r\}$ se satisface para toda selección X de $\widehat{X}$. La contradicción se encuentra con el hecho de que se puede obtener una selección X de $\widehat{X}$ satisfaciendo $Pl_{\widehat{X}}(\mathbb{B}_p(r)) = \mathbb{P}\{X \in \mathbb{B}_p(r)\}$, y para ello basta definir X como sigue: $X(w) \in \arg\min d_H(\widehat{X}(w), p)$, donde d_H denota la distancia Hausdorff (que X es punto aleatorio se deduce inmediatamente a partir de la propiedad de que $\widehat{X}$ es discreto).

La última afirmación de este teorema se deduce inmediatamente de la anterior.

El argumento entonces es claro: el/la decisor/a está eligiendo el conjunto aleatorio con el que es posible extraer un modelo exacto con menor varianza. Este es un argumento de peso que soporta la técnica que hemos introducido.

Ejemplo 1. *Consideremos* $\mathbb{R}$ *dotado de su distancia usual* d*. Consideremos los conjuntos aleatorios* $\widehat{X} = [X - r, X + r]$ *e* $\widehat{Y} = [Y - s, Y + s]$*, donde* X *está normalmente distribuida con media* μ_x *y desviación típica* σ_x*,* Y *también está normalmente distribuida con media* μ_y *y desviación típica* σ_y *y* r *es una constante estrictamente mayor que cero. Podemos interpretar estos conjuntos aleatorios como modelos exactos de probabilidad a los que se les añade un error o incertidumbre de valor* r*. Es fácil probar que* $\widehat{X}$ *es preferido a* $\widehat{Y}$ *si* $\sigma_x \leq \sigma_y$*. Obsérvese que esta decisión está en consonancia con el teorema anterior, dado que en este caso seremos capaces de extraer una selección, un modelo exacto, de* $\widehat{X}$ *con menor varianza que cualquier selección de* $\widehat{Y}$*.*

III. Conclusiones

En general, un problema de decisión en un ambiente impreciso acarrea una dificultad extra respecto a los problemas clásicos de teoría de la probabilidad. En esta contribución hemos introducido un nuevo orden dispersivo para conjuntos aleatorios. Este orden está basado en comparar las funciones de plausibilidad sobre la familia de bolas centradas en un punto.

Sería interesante seguir estudiando las propiedades que presenta este nuevo orden, y cómo simplificar el proceso de decisión. En este sentido, cabe estudiar qué particularidades presenta el orden para los casos en los que el espacio métrico sea bien finito o bien tiene la estructura de un espacio vectorial o afín euclídeo.

Por otro lado, también cabe extender este orden para otros modelos destacados en Probabilidades Imprecisas, como las variables aleatorias fuzzy.

Referencias

[1] I. Couso and D. Dubois. Statistical reasoning with set-valued information: Ontic vs. epistemic views. *International Journal of Approximate Reasoning*, pages 1502–1518, 2014.

[2] A. Giovagnoli and H.P. Wynn. Multivariate dispersion orderings. *Statistics & probability letters*, pages 325–332, 1995.

[3] I. Molchanov. Springer, 2005.

[4] A. Müller and D. Stoyan. *Comparison Methods for Stochastic Models and Risks*. Wiley, 2002.

[5] J.J. Salamanca. A universal, canonical dispersive ordering in metric spaces. *Journal of Statistical Planning and Inference*, 212:1–13, 2021.

[6] M. Shaked and J.G. Shanthikumar. *Stochastic orders*. Springer Science & Business Media, 2007.

[7] P. Walley. *Statistical reasoning with imprecise probabilities*. Chapman and Hall, 1991.

A fuzzy probability logic for compound conditionals

Tommaso Flaminio
IIIA – CSIC
08193 Bellaterra, Spain
ORCID 0000-0002-9180-7808
tommaso@iiia.csic.es

Lluis Godo
IIIA – CSIC
08193 Bellaterra, Spain
ORCID 0000-0002-6929-3126
godo@iiia.csic.es

***Abstract*—In this paper we propose a fuzzy modal logic for conditional probability that allows to represent and reason about the probability of not only basic conditional expressions of the form "φ given ψ", written $(\varphi \mid \psi)$, but also compound conditional sentences such as "φ given ψ *and* γ given χ", written $(\varphi \mid \psi) \wedge (\gamma \mid \chi)$, and more in general, any Boolean combination of basic ones. In order to formalize compound conditional formulas we will adopt the recently defined *Logic for Boolean Conditionals* (LBC) and hence formalize conditional probability as a simple (unconditional) probability of conditional sentences. In addition to such basic fuzzy modal logic for the probability of compound conditionals, we will also present some extensions and prove that each of them is sound and complete w.r.t. to a suitable class of probabilistic models. Furthermore, we will prove how to recover the usual interpretation of conditional probability, showing that, under minimal requirements, in these logics the probability of a basic conditional $(\varphi \mid \psi)$ can be safely taken as the conditional probability of φ given ψ, i.e. as the ratio $P(\varphi \wedge \psi)/P(\psi)$.**

***Index Terms*—Conditional probability; Compound conditional; Fuzzy logic; Fuzzy modal logic.**

I. Introduction

Fuzzy sets-based models and numerical uncertainty models, although sharing the feature of evaluating sentences in a totally ordered scale, usually the real unit interval $[0, 1]$, account for radically different notions of gradualness. From a formal point of view, these differences can be easily grasped if we consider their corresponding logics: fuzzy logics and uncertainty logics (in particular probability logics), respectively. In fact, while the former are truth-functional, i.e. the truth-value of a compound formula like $\varphi \vee \psi$ only depends on the truth-values of its components φ and ψ, the latter are not, since, for instance, the probability of $\varphi \vee \psi$ cannot be computed only from the probability of φ and the probability of ψ (it is also needed to know what is the probability of $\varphi \wedge \psi$).

Despite these differences, however, probability logics can be properly handled in a fuzzy logical setting by expanding the language of a fuzzy logic with a unary modality $P(\cdot)$ and interpreting, for every classical formula φ, the modal formula $P(\varphi)$ as "φ is probable". Clearly, $P(\varphi)$ is a fuzzy proposition, whose truth-degree can be taken as the probability of φ. More precisely, the fuzzy modal logic FP(Ł), as firstly introduced in [11] and improved in [10], extends the language of Łukasiewicz logic Ł by the modal operator $P(\cdot)$ and uses the ground logic Ł to express the basic properties of a probability function. In particular, it is worth to recall that the finite additivity of P can be expressed in FP(Ł) by using the Łukasiewicz connective $\oplus$ whose standard interpretation is the truncated sum: for all $x, y \in [0, 1]$, $x \oplus y = \min\{1, x+y\}$. Very recently, in [1] the authors have studied in depth the relationship of this fuzzy logic-based approach to more traditional probability logics after Halpern et al. see e.g. [13].

In addition to simple probability, the paper [8] presents the logic FP(ŁΠ) to deal with conditional probability by considering, instead of Ł, the stronger logic ŁΠ. Such formalism can be roughly regarded as the expansion of Łukasiewicz logic by the connectives of product conjunction $\odot$ and product implication $\to_\Pi$. The standard semantics of $\odot$ and $\to_\Pi$ interprets them, respectively, by the usual product $\cdot$ and the function $x \to_\Pi y = 1$ if $x \leq y$ and $x \to_\Pi y = y/x$ otherwise. Thus, if $P(\psi)$ is not zero, the conditional probability $P(\varphi \mid \psi)$ can be written in FP(ŁΠ) as $P(\psi) \to_\Pi P(\varphi \wedge \psi)$ and hence interpreted in its semantics as $P(\varphi \wedge \psi)/P(\psi)$. A related approach can be found in [9], where Popper conditional probabilities are formalised in a similar setting.

In this paper we propose a fuzzy modal logic FP(LBC, ŁΠ) for conditional probability that extends FP(ŁΠ) in the expressive power. In particular, FP(LBC, ŁΠ) formalizes conditional events by the recently defined logic LBC (Logic of Boolean Conditionals) for conditional events. The latter allows to represent not only basic conditional expressions "φ given ψ", written $(\varphi \mid \psi)$, but also compound conditional sentences such as "φ given ψ *and* γ given χ", written in LBC as $(\varphi \mid \psi) \wedge (\gamma \mid \chi)$, or more in general, any Boolean combination of basic ones [5]. For each of such (basic and compound) conditional sentences, FP(LBC, ŁΠ) permits to represent and reason about their probability. Thus, the conditional probability of "φ given ψ" is treated in FP(LBC, ŁΠ) as the *unconditional* probability of the basic conditional formula $(\varphi \mid \psi)$.

In addition, we will present extensions of FP(LBC, ŁΠ) that capture a more refined notion of probability functions. For FP(LBC, ŁΠ) and each of its extensions, we prove soundness and completeness results w.r.t. suitable classes of probability models.

This paper is organized as follows. Section II gathers extensive preliminaries: on the Logic for Boolean Conditionals (LBC) in Subsection II-A; on the ground propositional logic ŁΠ in Subsection II-B; and on the fuzzy modal logic for conditional probability FP(ŁΠ) in Subsection II-C. In Section

III we will define the probability logic for compound conditionals FP(LBC, ŁΠ). In the same section we will consider the class of *separable* models and prove completeness. Moreover, we will show that, for every basic conditional $(\varphi \mid \psi)$ such that ψ has positive probability, the logic FP(LBC, ŁΠ) proves that the modal formula $P(\varphi \mid \psi)$ is equivalent to $P(\psi) \to_\Pi P(\varphi \wedge \psi)$ and hence, in every separable model, $P(\varphi \mid \psi)$ is interpreted as the ratio $\mu(\varphi \wedge \psi)/\mu(\psi)$. A first extension of FP(LBC, ŁΠ), namely the logic FP(LBC, ŁΠ)$^+$, will be defined in Section IV and there we will show it to be complete w.r.t. to a subclass of separable models called *positive* separable models. Section V deals with the logic FP(LBC, ŁΠ)^+_c, a further extension meant to capture the behavior of the so-called *canonical* extensions to $\mathcal{C}(\mathbf{A})$ of positive (unconditional) probabilities on $\mathbf{A}$ in [5]. For FP(LBC, ŁΠ)^+_c we prove soundness and completeness w.r.t. to the proper subclass of positive separable models that will be called *canonical* in that section. Conclusions and future work on this subject will be discussed in the final Section VI.

II. Preliminaries

A. The logic LBC

In this section we recall from [5] the Logic of Boolean Conditionals (LBC). The idea is to consider basic conditional formulas of the form $(\varphi \mid \psi)$ as primitive objects that can be freely combined with Boolean connectives. A difference with the so-called *measure-free conditionals* is that the combination of two basic conditionals need not be another basic conditional, only in some special cases specified in the axioms of the logic. Indeed, formulas of LBC correspond to Boolean combinations of basic conditional formulas $(\varphi \mid \psi)$, where φ, ψ are classical propositions. In more detail, let L be a propositional language built from a finite set of propositional variables $p_1, p_2, \ldots, p_k$ and classical logic connectives $\wedge, \vee, \neg, \to, \leftrightarrow$. We will denote by $\vdash_{CPL}$ derivability in Classical Propositional Logic. Based on L, we define the language LBC of conditionals by the following stipulations:

- Basic (or atomic) conditional formulas, expressions of the form $(\varphi \mid \psi)$ where $\varphi, \psi \in \mathsf{L}$ and such that $\nvdash_{CPL} \neg\psi$, are in LBC.
- Further, if $\Phi, \Psi \in \mathsf{LBC}$, then $\neg\Phi, \Phi \wedge \Psi \in \mathsf{LBC}$.[1] Other connectives like $\vee$, $\to$ and $\leftrightarrow$ are defined as usual.

Note that we do not allow the nesting of conditionals, as usually done in the vast literature on the modal approaches to Conditional Logics Actually, purely propositional formulas from L can also be considered to be part of LBC since, as a matter of fact, any proposition φ can be identified with the conditional $(\varphi \mid \top)$, where $\top$ is an abbreviation for $\psi \vee \neg\psi$.

Definition 2.1: The *Logic of Boolean conditionals* (LBC for short) has the following axioms:

[1] We use the same symbols for connectives in L and in LBC without danger of confusion.

(CPL) For any tautology of CPL, the formula resulting from a uniform replacement of the variables by basic conditionals.
(A1) $(\psi \mid \psi)$
(A2) $\neg(\varphi \mid \psi) \leftrightarrow (\neg\varphi \mid \psi)$
(A3) $(\varphi \mid \psi) \wedge (\delta \mid \psi) \leftrightarrow (\varphi \wedge \delta \mid \psi)$
(A4) $(\varphi \mid \psi) \leftrightarrow (\varphi \wedge \psi \mid \psi)$
(A5) $(\varphi \mid \psi) \leftrightarrow (\varphi \mid \chi) \wedge (\chi \mid \psi)$, if $\vdash_{CPL} \varphi \to \chi$ and $\vdash_{CPL} \chi \to \psi$
(R1) from $\vdash_{CPL} \varphi \to \psi$ derive $(\varphi \mid \chi) \to (\psi \mid \chi)$
(R2) from $\vdash_{CPL} \chi \leftrightarrow \psi$ derive $(\varphi \mid \chi) \leftrightarrow (\varphi \mid \psi)$
(MP) Modus Ponens: from Φ and $\Phi \to \Psi$ derive Ψ

The notion of proof in LBC, $\vdash_{LBC}$, is defined as usual from the above axioms and rules.

In [5] it is shown that the Lindenbaum algebra corresponding to LBC, that is, the algebra of LBC-formulas modulo logical equivalence, is a certain type of Boolean algebra, called Boolean algebra of conditionals, which is finite if the set propositional variables is so, that is our case. Then, the algebra is atomic and its set of atoms are conjunctions of basic conditionals of length $n-1$, where $n = 2^m$ with m being the number of propositional variables, of the following form:

$$(\alpha_1 \mid \top) \wedge (\alpha_2 \mid \neg\alpha_1) \wedge \ldots \wedge (\alpha_{n_1} \mid \wedge_{i=1,n-2}\, \alpha_i),$$

where $\alpha_1, \ldots, \alpha_{n-1}$ are propositional atoms of the Lindenbaum algebra of the underlying propositional language L.

The semantics of the LBC logic is based on sequences $\overline{w} = \langle w_1, \ldots, w_n \rangle$ of pairwise different propositional interpretations for the underlying language L, in such a way that $\overline{w}$ satisfies a conditional $(\varphi \mid \psi)$, written $\overline{w} \models_{LBC} (\varphi \mid \psi)$, if $w_i \models_{CPL} \varphi$ for the lowest index i such that $w_i \models \psi$.

B. The logic ŁΠ

The ŁΠ is a powerful fuzzy logic system that suitably combines the connectives from Łukasiewicz logic with the connectives of Product fuzzy logic [4]. The language of the ŁΠ logic is built in the usual way from a countable set of propositional variables, three binary connectives $\to_L$ (Łukasiewicz implication), $\odot$ (Product conjunction) and $\to_\Pi$ (Product implication), and the truth constant $\bar{0}$. A truth-evaluation is a mapping e that assigns to every propositional variable a real number from the unit interval [0, 1] and extends to all formulas as follows:

$$\begin{aligned} &e(\bar{0}) = 0, \qquad e(\varphi \odot \psi) = e(\varphi) \cdot e(\psi) \\ &e(\varphi \to_L \psi) = \min(1 - e(\varphi) + e(\psi), 1), \\ &e(\varphi \to_\Pi \psi) = \begin{cases} 1, & \text{if } e(\varphi) \le e(\psi) \\ e(\psi)/e(\varphi), & \text{otherwise} \end{cases}. \end{aligned}$$

The truth constant $\bar{1}$ is defined as $\varphi \to_L \varphi$. In this way we have $e(\bar{1}) = 1$ for any truth-evaluation e. Moreover, many other connectives can be defined from those introduced above:

$$\begin{aligned} \neg_L\varphi \quad &\text{is} \quad \varphi \to_L \bar{0} \\ \neg_\Pi\varphi \quad &\text{is} \quad \varphi \to_\Pi \bar{0}, \\ \varphi \wedge \psi \quad &\text{is} \quad \varphi\&(\varphi \to_L \psi), \\ \varphi \vee \psi \quad &\text{is} \quad \neg_L(\neg_L\varphi \wedge \neg_L\psi), \end{aligned}$$

$$\begin{array}{rcl} \varphi \oplus \psi & \text{is} & \neg_L \varphi \to_L \psi, \\ \varphi \otimes \psi & \text{is} & \neg_L(\neg_L \varphi \oplus \neg_L \psi), \\ \varphi \ominus \psi & \text{is} & \varphi \& \neg_L \psi, \\ \varphi \leftrightarrow_L \psi & \text{is} & (\varphi \to_L \psi) \& (\psi \to_L \varphi), \\ \Delta\varphi & \text{is} & \neg_\Pi \neg_L \varphi, \\ \nabla\varphi & \text{is} & \neg_\Pi \neg_\Pi \varphi, \end{array}$$

with the following interpretations:

$$\begin{array}{rcl} e(\neg_L \varphi) & = & 1 - e(\varphi), \\ e(\neg_\Pi \varphi) & = & \begin{cases} 1, & \text{if } e(\varphi) = 0 \\ 0, & \text{otherwise} \end{cases}, \\ e(\varphi \wedge \psi) & = & \min(e(\varphi), e(\psi)), \\ e(\varphi \vee \psi) & = & \max(e(\varphi), e(\psi)), \\ e(\varphi \oplus \psi) & = & \min(1, e(\varphi) + e(\psi)), \\ e(\varphi \otimes \psi) & = & \max(0, e(\varphi) + e(\psi) - 1), \\ e(\varphi \ominus \psi) & = & \max(0, e(\varphi) - e(\psi)), \\ e(\varphi \leftrightarrow_L \psi) & = & 1 - |e(\varphi) - e(\psi)|, \\ e(\Delta\varphi) & = & \begin{cases} 1, & \text{if } e(\varphi) = 1 \\ 0, & \text{otherwise} \end{cases}, \\ e(\nabla\varphi) & = & \begin{cases} 1, & \text{if } e(\varphi) > 0 \\ 0, & \text{otherwise} \end{cases}. \end{array}$$

The logic ŁΠ is defined Hilbert-style as the logical system whose axioms and rules are the following:[2]

(i) Axioms of Łukasiewicz Logic:
(ii) Axioms of Product Logic
(iii) The following additional axioms relating Łukasiewicz and Product logic connectives:
  ($\neg$) $\neg_\Pi \varphi \to_L \neg_L \varphi$
  (Δ) $\Delta(\varphi \to_L \psi) \equiv \Delta(\varphi \to_\Pi \psi)$
  ($L\Pi$) $\varphi \odot (\psi \ominus \chi) \equiv (\varphi \odot \psi) \ominus (\varphi \odot \chi)$
(iv) Deduction rules of ŁΠ are modus ponens for $\to_L$ (modus ponens for $\to_\Pi$ is derivable), and necessitation for Δ: from φ derive $\Delta\varphi$.

C. The probability logic FP(ŁΠ)

This final section on preliminaries is on the probability logic FP(ŁΠ) (FP for Fuzzy Probability), introduced in [8] and defined as a sort of modal extension with a unary operator $P(\cdot)$ over the fuzzy logic ŁΠ described in the previous section. This logic allows for reasoning about the probability of classical propositions.

Actually, the propositional language L is extended by a fuzzy unary modal operator P. If φ is a proposition of L, then $P\varphi$ is a modal proposition whose intended reading is that "φ is *probable*", and whose truth-degree will be taken as the probability of φ.

The language of FP(ŁΠ) is defined as follows. *Formulas* of FP(ŁΠ) are of two types:

- Non-modal: they are exactly the (classical) formulas of L, i.e. those built from a set Var of propositional variables $\{p_1, p_2, \ldots p_n, \ldots\}$ using the classical binary connectives $\wedge$ and $\neg$. Other connectives like $\vee$ and $\to$ are defined from $\wedge$ and $\neg$ in the usual way. We shall denote them by lower case Greek letters φ, ψ, etc.
- Modal: they are built from elementary modal formulas of the form $P\varphi$, where φ is a non-modal formula, using the connectives of ŁΠ ($\to_L$, $\odot$, $\to_\Pi$). We shall denote them by upper case Greek letters Φ, Ψ, etc.

These are all the formulas of FP(ŁΠ). Notice that nested modalities, among other things, are not allowed.

Axioms and rules of FP(LBC, ŁΠ) are as follows:

(CPL) All axioms and rules of classical propositional logic restricted to classical, non-modal, formulas;
(ŁΠ) All axioms and rules of ŁΠ for modal formulas;
(P) The following axioms and rules for the modality P: for all propositions $\varphi, \psi \in \mathsf{L}$,
  (P1) $P(\varphi \to \psi) \to_L (P(\varphi) \to_L P(\psi))$
  (P2) $P(\neg\varphi) \leftrightarrow_L \neg P(\varphi)$
  (P3) $P(\varphi \vee \psi) \leftrightarrow_L [P(\varphi) \oplus (P(\psi) \ominus P(\varphi \wedge \psi))]$
  (Nec) if $\vdash_{CPL} \varphi$, derive $P(\varphi)$.

Models of FP(ŁΠ) are probability Kripke structures $K = \langle W, e, \mu \rangle$, where:

- W is a non-empty set of possible worlds;
- $e : V \times W \to \{0, 1\}$ provides for each world a *Boolean* (two-valued) evaluation of the proposition variables, that is, $e(p, w) \in \{0, 1\}$ for each propositional variable $p \in Var$ and each world $w \in W$; and
- $\mu : 2^W \to [0, 1]$ is a finitely additive probability measure on a Boolean algebra of subsets of W such that for each p, the set $\{w \mid e(p, w) = 1\}$ is measurable (cf. [10] 8.4.1).

A truth evaluation e is extended to non-modal formulas in the classical way, to elementary modal formulas as follows:

$$e(P\varphi, w) = \mu(\{w \in W \mid e(\varphi, w) = 1\}),$$

and to compound modal formulas by using the truth-functions of the ŁΠ logic.

Soundness and completeness of the logic FP(ŁΠ) w.r.t. to the class of probability Kripke models is proved in [8]: if $T \cup \{\Phi\}$ is a finite set of FP(ŁΠ)-formulas, then T proves Φ in FP(ŁΠ) iff for any probability Kripke model $K = (W, e, \mu)$ and any world $w \in W$, $e(\Phi, w) = 1$ whenever $e(\Psi, w) = 1$ for all $\Psi \in T$.

III. Probability logic over conditionals

In this section we define a logic to reason about the probability of basic and compound conditionals over the fuzzy logic ŁΠ. In the same line as with the logic FP(ŁΠ) described in Section II-C, we extend the language of LBC with a fuzzy (unary) modal operator P, so that, for every basic conditional $(\varphi \mid \psi)$, the intended meaning of a formula $P(\varphi \mid \psi)$ is that the conditional "φ given ψ" is *probable*, and that the truth-degree of $P(\varphi \mid \psi)$ is the probability of the conditional "$(\varphi \mid \psi)$". The relation of this probability to the usual notion of conditional probability of φ given ψ will become clear later.

The logic FP(LBC, ŁΠ) is obtained by replacing, in the definition of FP(ŁΠ), classical logic for events by the conditional logic LBC defined as in Section II-A. Formulas of FP(LBC, ŁΠ) are of two types:

[2]This definition, proposed in [3], is actually a simplified version of the original definition of $L\Pi$ given in [4].

- **Conditional formulas** are formulas of the logic LBC, that is, basic conditionals of the form $(\varphi \mid \psi)$ for all classical formulas φ and ψ such that ψ is not a classical logic contradiction (in other words, $\nvdash_{CPL} \neg\psi$) and compound conditional formulas obtained as Boolean combinations of basic ones. Compound conditional formulas will be denoted as $\Phi, \Psi, \ldots$;

- **Modal formulas**: for every (basic or compound) conditional formula Φ, $P(\Phi)$ is an atomic modal formula. Compound modal formulas are combinations of atomic ones by means of the ŁΠ connectives.

Thus, for instance, $P(\varphi \mid \psi)$, $P((\varphi \mid \psi) \wedge (\gamma \mid \delta))$ and $P((\varphi \mid \psi) \wedge (\gamma \mid \delta)) \to_L P(\chi \mid \tau)$ are compound modal formulas for all classical formulas $\varphi, \psi, \gamma, \delta, \chi, \tau$ such that $\nvdash \neg\psi$, $\nvdash \neg\delta$, $\nvdash \neg\tau$. However, neither $(\varphi \mid \psi) \to_L P(\gamma \mid \delta)$ nor $P((\varphi \mid \psi) \oplus P(\chi \mid \tau))$ are well formed formulas in this language.

Axioms and rules of FP(LBC, ŁΠ) are as follows:

(LBC) All axioms and rules of LBC restricted to conditional formulas;

(ŁΠ) All axioms and rules of ŁΠ for modal formulas;

(P) The axioms and rules for the modality P are those for FP(ŁΠ), but now for all conditional formulas $\Phi, \Psi \in$ LBC, plus a new rule (Sep):

(P1) $P(\Phi \to \Psi) \to_L (P(\Phi) \to_L P(\Psi))$;

(P2) $P(\neg\Phi) \leftrightarrow_L \neg P(\Phi)$;

(P3) $P(\Phi \vee \Psi) \leftrightarrow_L [P(\Phi) \oplus (P(\Psi) \ominus P(\Phi \wedge \Psi))]$;

(Nec) if $\vdash_{LBC} \Phi$, derive $P(\Phi)$;

(Sep) if $\vdash_{CPL} (\varphi \to \chi) \wedge (\chi \to \psi)$, derive $P((\varphi \mid \chi) \wedge (\chi \mid \psi)) \leftrightarrow_L P(\varphi \mid \chi) \odot P(\chi \mid \psi)$.

The notion of proof according to these axioms and rules will be denoted $\vdash_{FP}$. The axioms and rules of FP(LBC, ŁΠ) are meant to capture the behavior of an unconditional, *separable* probability measure on the Lindenbaum algebra of the logic LBC. In fact, let us recall from [5] that, in particular, a probability μ on a Boolean algebra of conditionals $\mathcal{C}(\mathbf{A})$ is *separable* if for all $a, b, c \in A \setminus \{\bot\}$ such that $a \leq b \leq c$, then $\mu((a \mid b) \wedge (b \mid c)) = \mu(a \mid b) \cdot \mu(b \mid c)$. As we will show later on, separability is captured by the rule (Sep) above.

In what follows Ω will denote the set of Boolean interpretations for the variables Var, and $Seq_n(\Omega)$ will denote the set of sequences of n pairwise different interpretations from Ω.

Definition 3.1: A probability LBC-Kripke model is a structure $\mathcal{C} = \langle W, e, \mu\rangle$ where

- W is a set of worlds;
- $e : W \to Seq_n(\Omega)$ maps every world $w \in W$ to a LBC-evaluation $e(w) = \overline{w} \in Seq_n(\Omega)$;
- μ is a probability on $2^{e[W]}$, where $e[W] = \{e(w) \mid w \in W\} \subseteq 2^{Seq_n(\Omega)}$.

Each LBC-Kripke model $\mathcal{C} = \langle W, e, \mu\rangle$ induces a probability on LBC formulas in the natural way, in particular for each basic conditional $(\varphi \mid \psi)$ we define:

$$\mu(\text{“}(\varphi \mid \psi)\text{”}) = \mu(\{\overline{w} \in Seq_n(\Omega) \mid w \in W, \overline{w} \models (\varphi \mid \psi)\},^3$$

[3] Without danger of confusion, we will write $\mu(\varphi \mid \psi)$ for $\mu(\text{“}(\varphi \mid \psi)\text{”})$.

and similarly for every compound conditional. Notice that, when $e[W] = Seq_n(\Omega)$, the Boolean algebra $2^{e[W]}$ on which μ is defined, actually is the Lindenbaum algebra of LBC. It was proved in [5, Theorem 7.3] that such Lindenbaum algebra is isomorphic to $\mathcal{C}(\mathbf{L})$, that is, the conditional algebra generated by $\mathbf{L}$, the Lindenbaum algebra of classical propositional logic. Thus, every LBC-Kripke model determines a probability measure μ on $\mathcal{C}(\mathbf{L})$.

Definition 3.2: A probability LBC-Kripke model $\mathcal{C} = \langle W, e, \mu\rangle$ is called *separable* when

$$\mu((\varphi \mid \psi) \wedge (\psi \mid \chi)) = \mu(\varphi \mid \psi) \cdot \mu(\psi \mid \chi) \qquad (1)$$

for every φ, ψ, χ such that $\vdash_{CPL} (\varphi \to \psi) \wedge (\psi \to \chi)$ and $\nvdash_{CPL} \neg\psi$.

An immediate consequence of the definition above is that every μ of a separable model, satisfies $\mu(\varphi \mid \top) = \mu((\varphi \mid \psi) \wedge (\psi \mid \top)) = \mu(\varphi \mid \psi) \cdot \mu(\psi \mid \top)$ for all φ, ψ such that $\vdash_{CPL} \varphi \to \psi$ and $\nvdash \neg\psi$. Note that the first equality is due to Axiom (A5) of LBC.

Given a formula F of FP(LBC, ŁΠ) and a separable model $\mathcal{S} = \langle W, e, \mu\rangle$, the evaluation of F in $\mathcal{S}$ at $w \in W$ is inductively defined by the following stipulations:

- If F is a conditional formula Φ, then $\|\Phi\|_{\mathcal{S},w} = w_1(\Phi) \in \{0, 1\}$, where $e(w) = \langle w_1, w_2, \ldots\rangle$;
- If $F = P(\Phi)$ is atomic modal, then $\|P(\Phi)\|_{\mathcal{S},w} = \mu(\Phi)$;
- If F is compound modal, then $\|F\|_{\mathcal{S},w}$ is computed by evaluating its atomic components and then by using the truth-functionality of the connectives of ŁΠ in the standard algebra $[0, 1]$.

Notice that if F is modal, then $\|F\|_{\mathcal{S},w}$ does not depend on the world w. Finally, the truth-degree of F in $\mathcal{C}$ is defined as $\|F\|_{\mathcal{S}} = \inf_{w \in W} \|F\|_{\mathcal{S},w}$.

Definition 3.3: If $T \cup \Phi$ is a set of modal formulas, Φ logically follows from T, written $T \models_{SFP} \Phi$, when for all separable probability LBC-Kripke model $\mathcal{S}$, if $\|F\|_{\mathcal{S}} = 1$ for every $F \in T$, then $\|\Phi\|_{\mathcal{S}} = 1$ as well.

Next, we prove that FP(LBC, ŁΠ) is sound and complete with respect to the class of separable models. Its proof, a main part of which will follow from a general result we will recall below, is based on the fact that the logic for events, LBC is locally finite. This means that the Lindenbaum algebra $\mathbf{CL}$ over a finite set of variables is finite. Indeed, as proved in [5, Theorem 7.3], $\mathbf{CL}$ is isomorphic to $\mathcal{C}(\mathbf{L})$ to the Boolean algebra of conditionals of the Lindenbaum algebra of classical logic, on the same set of variables.

Theorem 3.1: The logic FP(LBC, ŁΠ) is sound and complete for deductions from probabilistic modal formulas w.r.t. to the class of separable models.

Proof: Soundness of (P1), (P2), (P3) and (Nec) follows directly from [10, Lemma 8.4.5.]. Thus, let us show that (Sep) holds in every separable model. If $(\varphi \to \chi) \wedge (\chi \to \psi)$ is a theorem of classical logic and ψ is not a contradiction, then $\|P((\varphi \mid \chi) \wedge (\chi \mid \psi))\|_{\mathcal{S}} = \|P(\varphi \mid \chi)\|_{\mathcal{S}} \cdot \|P(\chi \mid \psi)\|_{\mathcal{S}}$ in any separable model $\mathcal{S} = \langle W, e, \mu\rangle$, because μ satisfies Equation (1) above, and hence $\mathcal{S}$ satisfies $P((\varphi \mid \chi) \wedge (\chi \mid \psi)) \leftrightarrow_L P(\varphi \mid \chi) \odot P(\chi \mid \psi)$.

As for completeness, we will show that $\nvdash_{FP} F$ implies $\not\models_{SFP} F$, for any modal formula F. By either adapting the completeness proofs in [8], [11] or adapting the general result proved in [6, Theorem 20 (1)], together with the fact that LBC is locally finite and ŁΠ is finitely strong standard complete, we can show that deductions in FP(LBC, ŁΠ) can be translated to deductions in ŁΠ by considering atomic modal formulas $P\Phi$ as new ŁΠ propositional variables p_Φ. Indeed, it holds that, for any modal formula G, $\vdash_{FP} G$ iff $T \vdash_{\text{ŁΠ}} G^*$, where G^* is the translation of G with the new variables, and T consists of the following three sets of formulas:

(i) $T_0 = \{H^* : H$ is an instance of axioms (P1), (P2), (P3)$\}$

(i)i $T_1 = \{p_\Psi : \psi$ is an LBC-theorem$\}$, that translates the rule (Nec), and

(iii) $T_2 = \{p_{(\varphi|\chi)\wedge(\chi|\psi)} \leftrightarrow_L p_{\varphi|\chi} \odot p_{\chi|\psi} : \vdash_{CPL} (\varphi \to \chi) \wedge (\chi \to \psi)$ and $\nvdash_{CPL} \neg\chi\}$, that translates the rule (Sep).

Then by the finite-strong completeness of ŁΠ,[4] if F is not a theorem of FP(LBC, ŁΠ), there is a ŁΠ-evaluation v, model of the sets T_0, T_1, T_2 and $v(F) < 1$. Then one can define the probability Kripke model $\mathcal{S} = (W, e, \mu)$, where $W = Seq_n(\Omega)$, $e(w, p) = \overline{w}_1(p)$ for any propositional variable p, and $\mu(\Phi) = v(\Phi)$ for any conditional formula Φ, and show that $\|F\|_{\mathcal{S}} = v(F) < 1$, that is, $\mathcal{S}$ is a countermodel of F. Thus, it is left to prove that μ is separable. Indeed, in particular v is a model of T_2, that means $v(p_{\varphi|\chi} \odot p_{\chi|\psi}) = v(p_{\varphi|\chi}) \cdot v(p_{\chi|\psi}) = \mu(\varphi \mid \chi) \cdot \mu(\chi \mid \psi)$, for all those conditionals $(\varphi \mid \chi), (\chi \mid \psi)$ such that $\vdash_{CPL} (\varphi \to \chi) \wedge (\chi \to \psi)$ and $\nvdash_{CPL} \neg\chi$. Thus, μ is separable and the claim is settled. ■

We end this section by noticing that in a separable LBC-Kripke model, a formula $P(\varphi \mid \psi)$ is evaluated by its corresponding conditional probability.

Corollary 3.1: For every basic conditional $(\varphi \mid \psi)$, the following deduction holds in FP(LBC, ŁΠ):[5]

$$\nabla P(\psi \mid \top) \vdash_{FP} P(\varphi \mid \psi) \leftrightarrow_L (P(\psi \mid \top) \to_\Pi P(\varphi \wedge \psi \mid \top))$$

IV. Positive separable models

In this section we will consider a first extension of the logic FP(LBC, ŁΠ) that allows to deal with positive probabilities on basic conditionals of the form $(\varphi \mid \top)$.

Definition 4.1: The logic FP(LBC, ŁΠ)$^+$ is the schematic extension of FP(LBC, ŁΠ) obtained by adding the rule

(Pos) if $\nvdash_{CPL} \neg\varphi$, derive $\nabla P(\varphi \mid \top)$.

The effect of axiom (Pos) is to force the probability of non-contradictory classical propositions φ (once identified as conditionals $(\varphi \mid \top)$) to be strictly positive. Therefore, it is relatively easy to see that the following holds.

Theorem 4.1: The logic FP(LBC, ŁΠ)$^+$ is sound and complete w.r.t. the class of *positive* separable LBC-Kripke models, i.e. models $\mathcal{S} = \langle W, e, \mu\rangle$ in which μ is a positive probability measure, that is to say, such that $\mu(\Phi) > 0$ for all conditional formula $\Phi \neq \bot$.

Let us call *basic modal formulas* any combination of atomic modal formulas of the form $P(\varphi_i \mid \psi_i)$ with ŁΠ connectives. If we restrict ourselves to this sublanguage of FP(LBC, ŁΠ), we can in fact consider simpler probabilistic models.

Definition 4.2: A *positive simple model* is a pair $\mathcal{P} = \langle \Omega, \sigma\rangle$ where Ω is the set of Boolean interpretations for the base language L, and σ is a positive probability measure on 2^Ω.

Given a basic modal formula B and a positive model $\mathcal{P} = \langle \Omega, \sigma\rangle$, we interpret B in $\mathcal{P}$ as follows:

- If $B = P(\varphi \mid \psi)$, then $\|P(\varphi \mid \psi)\|_{\mathcal{P}} = \frac{\sigma(\varphi\wedge\psi)}{\sigma(\psi)}$;
- If B is compound use again the truth functionality of ŁΠ connectives interpreted in $[0, 1]$.

Theorem 4.2: (1) For every positive separable LBC-Kripke model $\mathcal{S}$ there exists a positive simple model $\mathcal{P}$ such that $\|B\|_{\mathcal{S}} = \|B\|_{\mathcal{P}}$ for every basic modal formula B.
(2) Vice-versa, for every positive simple model $\mathcal{P}$ there exists a positive separable LBC-Kripke model $\mathcal{S}$ such that $\|B\|_{\mathcal{P}} = \|B\|_{\mathcal{S}}$ for every basic modal formula B.

Proof: As for (1), let us prove the clam for $B = (\varphi \mid \psi)$. The case of compound conditional formulas, indeed, follows by truth-functionality of the connectives of ŁΠ. Given a positive separable LBC-Kripke model $\mathcal{S} = \langle W, e, \mu\rangle$, define $\sigma(\varphi) = \mu(\varphi \mid \top) = \mu(\{w \in W \mid \overline{w} \models (\varphi \mid \top)\}$. This is a probability on Boolean formulas that can be identified as a probability on 2^Ω. Since μ is positive and separable, we have

$$\mu(\varphi \mid \psi) = \frac{\mu(\varphi \wedge \psi \mid \top)}{\mu(\psi \mid \top)} = \frac{\sigma(\varphi \wedge \psi)}{\sigma(\psi)}.$$

On the other hand, (2) follows by adapting to our logical setting a main result in [5, Theorem 6.13] stated in algebraic terms. Indeed, in that theorem it is proved that, for any positive probability P on an algebra of events $\mathbf{A}$, there is a (plain) probability μ_P on the algebra of conditional events $\mathcal{C}(\mathbf{A})$ such that $\mu_P(a \mid b) = \frac{\mu(a\wedge b)}{\mu(b)}$ whenever $b \neq \bot$. The proof is rather involved and we refer the reader to [5] for full details. ■

V. A logic for (conditional) canonical extensions

As proved in [5] the atoms of a Boolean algebra of conditionals $\mathcal{C}(\mathbf{A})$ can be fully characterized by the atoms of the original algebra $\mathbf{A}$ and, in particular, if $\alpha_1, \ldots, \alpha_n$ are the atoms of $\mathbf{A}$, those of $\mathcal{C}(\mathbf{A})$ are conditional expressions of the form

$$\omega_i = (\alpha_{i_1} \mid \top) \wedge (\alpha_{i_2} \mid \neg\alpha_{i_1}) \wedge \ldots \wedge (\alpha_{i_{n-1}} \mid \bigwedge_{j\leq n-2} \neg\alpha_{i_j}).$$

Since the atoms of the Lindenbaum algebra of classical logic (with, say, k variables $x_1, \ldots, x_k$) are writable as minterms $\alpha_j = \bigwedge x_i^*$, for $x_i^* \in \{x_i, \neg x_i\}$, the atoms of $\mathcal{C}(\mathbf{L})$ are expressible as above. To ease the reading, we will denote them by $\omega_1, \omega_2, \ldots$. Recall form [5] that, if classical logic is defined on k propositional variables, there are $(2^k)!$ atoms of $\mathcal{C}(\mathbf{L})$.

Although not every probability measure on $\mathcal{C}(\mathbf{A})$ satisfies all the axioms of a conditional probability, every positive probability σ on the original algebra $\mathbf{A}$, has an extension to

[4] For our purposes, T can be considered to be finite, because the Lindenbaum Boolean algebra of LBC is finite, that is, there are only finitely-many non-logically equivalent basic conditionals $(\varphi \mid \psi)$, and hence only finitely-many non-logically equivalent formulas in T.

[5] Recall from Section II-B that the interpretation of the connective ∇ is $\nabla(x) = 1$ if $x > 0$, and $\nabla(x) = 0$ otherwise.

a positive probability μ_σ on $\mathcal{C}(\mathbf{A})$. These measures, called *canonical* in [5] are such that, for every atom ω_i of $\mathcal{C}(\mathbf{L})$,

$$\mu_\sigma(\omega_i) = \sigma(\alpha_{i_1} \mid \top)\cdot\sigma(\alpha_{i_2} \mid \neg\alpha_{i_1})\cdot\ldots\cdot\sigma(\alpha_{i_{n-1}} \mid \bigwedge_{j\leq n-2} \neg\alpha_{i_j}).$$

In this section we will show how to further extend the logic $\mathrm{FP}(\mathrm{LBC}, \text{Ł}\Pi)^+$ in order for its models to be defined by canonical extensions μ_σ of this kind. In order to do that, let us consider the following $\mathrm{FP}(\mathrm{LBC}, \text{Ł}\Pi)^+$-formulas:

$$(\mathrm{Can}_i) \quad P(\omega_i) \leftrightarrow_L P(\alpha_{i_1} \mid \top)\odot\ldots\odot P(\alpha_{i_{n-1}} \mid \bigwedge_{j\leq n-2} \neg\alpha_{i_j}),$$

where $\omega_i = (\alpha_{i_1} \mid \top) \wedge \ldots \wedge (\alpha_{i_{n-1}} \mid \bigwedge_{j\leq n-2} \neg\alpha_{i_j})$, with $\alpha_{i_1}, \ldots, \alpha_{i_n}$ being minterms of the propositional language L.

Definition 5.1: Let L be a propositional language with k variables. Then, the logic $\mathrm{FP}(\mathrm{LBC}, \text{Ł}\Pi)^+_c$ is the schematic extension of $\mathrm{FP}(\mathrm{LBC}, \text{Ł}\Pi)^+$ obtained by adding the axioms (Can_i) for all $i = 1, \ldots, (2^k)$!

A separable model $\mathcal{S} = \langle W, e, \mu\rangle$ is *canonical* if there exists a positive probability σ on Ω such that $\mu = \mu_\sigma$, i.e., μ is the canonical extension of some positive σ on Ω.

Finally, we can prove that $\mathrm{FP}(\mathrm{LBC}, \text{Ł}\Pi)^+_c$ is sound and complete w.r.t. to canonical models.

Theorem 5.1: The logic $\mathrm{FP}(\mathrm{LBC}, \text{Ł}\Pi)^+_c$ is sound and complete with respect to the class of canonical models.

Proof: Following the lines we sketched in the proof of Theorem 3.1, it is enough to show that a positive separable model satisfies all the axioms (Can_i) iff the model is canonical.

(Left-to-Right). Let $\mathcal{S} = \langle W, e, \mu\rangle$ be positive separable and satisfying (Can_i) for all i. Thus, $\|P(\omega_i)\|_\mathcal{S} =$
$\|P(\alpha_{i_1} \mid \top) \odot \ldots \odot P(\alpha_{i_{n-1}} \mid \bigwedge_{j\leq i_{n-2}} \neg\alpha_j)\|_\mathcal{S} =$
$\|P(\alpha_{i_1} \mid \top)\|_\mathcal{S} \cdot \ldots \cdot \|P(\alpha_{i_{n-1}} \mid \bigwedge_{j\leq i_{n-2}} \neg\alpha_j)\|_\mathcal{S} =$
$\mu(\alpha_{i_1} \mid \top) \cdot ldots \cdot \mu(\alpha_{i_{n-1}} \mid \bigwedge_{j\leq i_{n-2}} \neg\alpha_j)$.

Since $\mathcal{S}$ is positive and separable, by Theorem 4.2 there is a positive simple model $\mathcal{P} = (\Omega, \sigma)$ such that, for every basic conditional $(\varphi \mid \psi)$, $\mu(\varphi \mid \psi) = \frac{\sigma(\varphi\wedge\psi)}{\sigma(\psi)}$. In particular, $\mu(\alpha_{i_1} \mid \top)\cdot\ldots\cdot\mu(\alpha_{i_{n-1}} \mid \bigwedge_{j\leq i_{n-2}} \neg\alpha_j) = \frac{\alpha_{i_1}}{1} \cdot \frac{\sigma(\alpha_{i_2}\wedge\neg\alpha_{i_1})}{\sigma(\neg\alpha_{i_1})} \cdot \ldots \cdot \frac{\sigma(\alpha_{i_{n-1}}\wedge\bigwedge_{j\leq i_{n-2}}\neg\alpha_j)}{\sigma(\bigwedge_{j\leq n-2}\neg\alpha_{i_j})}$. Thus, $\mu = \mu_\sigma$ and $\mathcal{S}$ is canonical.

(Right-to-Left). Conversely, if $\mathcal{S}$ is canonical, then (Can_i) holds in $\mathcal{S}$ by the very definition of canonical model and the way formulas are interpreted in separable models. ■

In the light of the above argument, we can hence slightly improve the result of Corollary 3.1 as follows.

Corollary 5.1: The following formulas are theorems of $\mathrm{FP}(\mathrm{LBC}, \text{Ł}\Pi)^+_c$:

1) $P(\varphi \mid \psi) \leftrightarrow_L (P(\psi \mid \top) \rightarrow_\Pi P(\varphi \wedge \psi \mid \top))$;

2) $P(\omega) \leftrightarrow_L [P(\top) \rightarrow_\Pi P(\alpha_{i_1}) \odot P(\neg\alpha_{i_1}) \rightarrow_\Pi P(\alpha_{i_2}) \odot \ldots \odot P(\bigwedge_{j\leq i_{n-2}} \alpha_j) \rightarrow_\Pi P(\alpha_{i_{n-1}})]$, for $\omega = (\alpha_{i_1} \mid \top) \wedge (\alpha_{i_2} \mid \neg\alpha_{i_1}) \wedge \ldots \wedge (\alpha_{i_{n-1}} \mid \bigwedge_{j\leq n-2} \alpha_{i_j})$, where the α_{i_j}'s are pairwise different minterms of the propositional language L.

VI. Conclusions

In this paper we have introduced fuzzy modal logics for reasoning about the probability of compound conditionals, the latter being Boolean combinations of basic conditionals $(\varphi \mid \psi)$ and formalized within the recently introduced Logic for Boolean Conditionals LBC [5]. For each of the logics we define, we have proved completeness w.r.t. suitable classes of probability models, where a formula of the kind $P(\varphi \mid \psi)$ is evaluated by a (plain) probability $\mu(\text{“}(\varphi \mid \psi)\text{”})$ of the conditional formula $(\varphi \mid \psi)$. We have shown that, if $\psi \nvdash_{CPL} \bot$, this probability is in fact a conditional probability, and thus evaluated by the ratio $\mu(\text{“}(\varphi \wedge \psi \mid \top)\text{”})/\mu(\text{“}(\psi \mid \top)\text{”})$.

There are a number of issues on this subject left for future work. Among them, we plan to investigate the relationship of our probability logics for compound conditionals with the approach developed by Sanfilippo et al. to probabilistic inference with conjoined and iterated conditionals based on a different notion of conditional, see e.g. [14], [15]. Another topic of interest is the application of these logics to reason with (semi-) fuzzy quantifiers [2]. In addition, we plan to investigate complexity bounds for the SAT problem for $\mathrm{FP}(\mathrm{LBC}, \text{Ł}\Pi)$. Although it seems reasonable to conjecture that logic to be decidable, while the logics ŁΠ and $\mathrm{FP}(\text{Ł}\Pi)$ are known to be in PSPACE [12], the complexity of the logic LBC is not known yet, and this latter non-trivial fact needs to be solved first.

Acknowledgments: The authors are grateful to anonymous reviewers for their comments. The authors acknowledge partial support by the Spanish project ISINC (PID2019-111544GB-C21). Flaminio also acknowledges partial support by the Spanish Ramón y Cajal research program RYC-2016-19799.

References

[1] P. Baldi, P. Cintula, C. Noguera. Classical and Fuzzy Two-Layered Modal Logics for Uncertainty: Translations and Proof-Theory. *Int. J. Comput. Intell. Syst.* 13(1): 988-1001, 2020.

[2] P. Baldi, C.G. Fermüller. From Semi-fuzzy to Fuzzy Quantifiers via Łukasiewicz Logic and Games. Proc. EUSFLAT/IWIFSGN, Vol. 1, 112-124, 2017.

[3] P. Cintula The ŁΠ and ŁΠ1/2 propositional and predicate logics. *Fuzzy Sets and Systems*, 124, pp. 289–302, 2001.

[4] F. Esteva, L. Godo and F. Montagna. The ŁΠ and ŁΠ1/2 logics: two complete fuzzy logics joining Lukasiewicz and product logic. *Archive for Mathematical Logic*, 40, 39-67, 2001.

[5] T. Flaminio, L. Godo, H. Hosni. Boolean algebras of conditionals, probability and logic. *Artificial Intelligence* 286, 103347, 2020.

[6] T. Flaminio, L. Godo, E. Marchioni. Reasoning about Uncertainty of Fuzzy Events: an Overview. In *Understanding Vagueness - Logical, Philosophical, and Linguistic Perspectives*, P. Cintula et al. (Eds.), College Publications, pp. 367–400, 2011.

[7] A Gilio, G. Sanfilippo. Generalized logical operations among conditional events. *Appl. Intell.* 49(1): 79-102, 2019.

[8] L. Godo, F. Esteva, P. Hájek. Reasoning about probability using fuzzy logic. *Neural Network World*, 10, No. 5, pp. 811–824, 2000.

[9] L. Godo, E. Marchioni. Coherent Conditional Probability in a Fuzzy Logic Setting. *Log. J. IGPL* 14(3): 457-481, 2006.

[10] P. Hájek. *Metamathematics of fuzzy logic,* Kluwer 1998.

[11] P. Hájek, L. Godo, F. Esteva, Probability and Fuzzy Logic. In *Proc. of Uncertainty in Artificial Intelligence* UAI'95, 1995.

[12] P. Hájek, S. Tulipani, Complexity of fuzzy probability logics. *Fundamenta Informaticae*, 45, 1–7, 2001.

[13] J. Halpern. *Reasoning about Uncertainty*. MIT Press, 2003.

[14] G. Sanfilippo, N. Pfeifer, D.E. Over, A. Gilio. Probabilistic inferences from conjoined to iterated conditionals. *Int. J. Approx. Reason.* 93: 103-118, 2018.

[15] G. Sanfilippo, A. Gilio, D.E. Over, N. Pfeifer. Probabilities of conditionals and previsions of iterated conditionals. *Int. J. Approx. Reason.* 121: 150-173, 2020.

Empleo de contextos L-fuzzy en el diseño de materiales de origen renovable

Itsaso Leceta
Dep. Matemática Aplicada
Universidad del País Vasco - UPV/EHU
San Sebastián
itsaso.leceta@ehu.eus

Cristina Alcalde
Dep. Matemática Aplicada
Universidad del País Vasco - UPV/EHU
San Sebastián
c.alcalde@ehu.eus

Marta Urdanpilleta
Dep. Física Aplicada
Universidad del País Vasco - UPV/EHU
San Sebastián
marta.urdanpilleta@ehu.eus

Pedro Guerrero
Dep. Ing. Química y Medio Ambiente
Universidad del País Vasco - UPV/EHU
San Sebastián
pedromanuel.guerrero@ehu.eus

Koro de la Caba
Dep. Ing. Química y Medio Ambiente
Universidad del País Vasco - UPV/EHU
San Sebastián
koro.delacaba@ehu.eus

Ana Burusco
Dep. Estadística, Infor. y Matemáticas
Universidad Pública de Navarra
Instituto de Smart Cities
Pamplona
burusco@unavarra.es

***Resumen*—El objetivo principal de este trabajo es utilizar los contextos L-fuzzy para mejorar la eficiencia del diseño de materiales biodegradables de origen renovable fabricados mediante la valorización de residuos con buenas propiedades para su utilización en el sector de los envases alimentarios. El uso de los contextos L-fuzzy permitirá mejorar la eficacia en el proceso de diseño mejorando la eficiencia en la utilización de recursos. Además, mediante la utilización de los mencionados contextos L-fuzzy se pretende optimizar las propiedades de los materiales desarrollados y estimar la composición de los materiales para los requisitos que cumplir en servicio.**

***Palabras Clave*—Contextos L-fuzzy, extracción de conocimiento, materiales renovables, diseño de materiales**

I. Introducción

Cuando se diseña un nuevo material, la selección de la formulación es uno de los principales factores que afecta a las propiedades funcionales del material. En el proceso usual del diseño de nuevos materiales, se varía la cantidad de los componentes de la formulación y se estudia la influencia de esas variaciones en las propiedades del material. Dicho proceso implica un elevado coste de materias primas, energía y recursos humanos. Cuando un/a ingeniero/a diseña un nuevo material, los requerimientos del material en servicio dependiendo de su aplicación determinan las propiedades funcionales que el material debe poseer. Habitualmente, existe un conocimiento existente por parte del equipo investigador que ayuda a establecer unos rangos iniciales de los valores de la formulación con el objetivo de obtener las propiedades deseadas. Muchas veces, el investigador no busca un valor exacto para una propiedad concreta, sino que desea que esté dentro de un intervalo determinado o entorno a un valor exacto. Los valores de esas propiedades pasan a ser difusas, lo que posibilita el estudio mediante variables borrosas [1], [2].

La inexistencia a priori de reglas de asociación que relacionen la composición con las diversas propiedades del material hace que el uso de herramientas difusas basadas en reglas de asociación no parezca la mejor opción, siendo el empleo de contextos borrosos para la extracción de información una herramienta prometedora para la estimación de la composición del material para unas propiedades determinadas. Las propiedades requeridas van cambiando en función de las aplicaciones. Es decir, según el alimento que se quiera envasar, los requerimientos en servicio del material de envasado varían. El deseo de obtener una modificación en alguna de las propiedades obtenidas o en alguna variable de la formulación conlleva un coste inmenso de recursos debido al proceso de prueba y error que se utiliza actualmente. El empleo de contextos borrosos con información incompleta puede ayudar en la fase de diseño del material agilizando el proceso y ahorrando recursos de todo tipo que dicho diseño conlleva.

Trabajo parcialmente subvencionado por el Gobierno Vasco (Proyecto KK-2019/00006)

II. Descripción del problema

II-A. Descripción de las condiciones de diseño de materiales

En el diseño de nuevos materiales, la formulación y las condiciones de procesamiento desempeñan un papel fundamental en el comportamiento en servicio de dichos materiales. Estas propiedades están relacionadas con distintas magnitudes (químicas, físicas, mecánicas, térmicas, ópticas…). Para cada aplicación, sabemos qué valores de salida de estas magnitudes debería tener el material para un desempeño óptimo, y la cuestión es encontrar la formulación que lo permita.

Con vistas a alcanzar este objetivo, no sólo se debe identificar las especies químicas que optimizan las propiedades, sino también sus cantidades. Estas cantidades serán los valores de entrada que se han de determinar en la optimización. Usualmente, existe un conocimiento previo que permite al/la diseñador/a fijar los rangos aproximados en donde es probable que estén los valores de entrada ideales, para obtener los valores de salida asociados al comportamiento que son los deseados.

A la hora de encontrar esos valores de entrada ideales, la aproximación tradicional de prueba y error puede requerir un gran coste de energía, tiempo y materias primas. Hay tantas combinaciones y variaciones a estudiar que éste es un procedimiento ineficiente para analizar la influencia de cada parámetro en el comportamiento del material diseñado. Además de esto, normalmente el análisis se realiza con una variable de cada vez, que no considera las interacciones cruzadas entre distintas variables de entrada en la formulación. A menudo, estas entradas y salidas son borrosas: el/la diseñador/a aceptaría valores de entrada alrededor de ciertos valores de entrada, porque estaría dispuesto/a a admitir valores salida cerca de ciertos valores de salida.

En este caso, les podemos asociar valores borrosos de deseabilidad, y realizar un estudio basado en los conceptos de la teoría borrosa. Dentro de esta aproximación, y con un análisis estricto de las variables implicadas, se puede lograr un conjunto de variables de entrada adecuadas, de una manera eficiente y reduciendo el consumo de tiempo y recursos. Así, se evita un procedimiento ineficiente y sus costos añadidos.

II-B. Propiedades requeridas para aplicaciones del material

Los films basados en gelatina muestran características que los hacen indicados para ser aplicados en envasado [3]–[5]. Las propiedades barrera, el comportamiento mecánico y las propiedades ópticas son de vital importancia al diseñar materiales para envasado de alimentos [6].

Teniendo esto en mente, se proponen las siguientes propiedades de salida para un análisis posterior:

- Ángulo de contacto (CA)
- Tasa de transmisión de vapor de agua (WVTR)
- L* y b* (en lo que respecta a propiedades de color)
- Tensión a tracción (TS)
- Elongación a rotura (EB)
- Brillo

Considerando la experiencia en el campo del grupo de investigación *BIOMAT - biopolymeric materials* del que forman parte varios de los autores del presente trabajo, se propuso un conjunto de propiedades salida objetivo para esta aplicación. Este set tiene en cuenta las especificidades de los films basados en gelatina. Este es el punto de partida para lanzar el análisis de conceptos L-fuzzy [7]–[9]. Estas propiedades objetivo están listadas en la Tabla I.

Tabla I
REQUERIMIENTOS PARA MATERIALES DE ENVASADO DE ALIMENTOS GRASOS

CA (°)	L*	b*	TS (MPa)	EB (%)	Brillo_{60° (Unidad de brillo)
90	95	40	60	5	50

En el caso de la tasa de transmisión de vapor de agua, no se seleccionaron valores objetivo por no ser una propiedad clave para la aplicación de envasado de comida grasa. Puede ser que las propiedades objetivo no se puedan alcanzar simultáneamente para un material dado, debido a la interacción entre ellos. En este caso, el método de estimación propuesto proporciona una solución de compromiso en el que las propiedades estimadas están cerca de las propiedades objetivo.

En la siguiente sección se discuten los conceptos básicos de la teoría de contextos L-fuzzy y la forma de aplicarlos a la selección de formulaciones de materiales.

III. MODELADO DE LOS EXPERIMENTOS UTILIZANDO CONTEXTOS L-FUZZY

III-A. Análisis de contextos L-fuzzy

El análisis formal de conceptos es una herramienta matemática desarrollada por R. Wille en 1982 [10], [11] con el objetivo de procesar datos conceptuales y representarlos de una manera formal. Con el objetivo de reconocer diferentes grados de pertenencia entre objetos y atributos Burusco y Fuentes-González extendieron los conceptos formales de Wille al caso difuso con la definición de contextos L-fuzzy [7]–[9]. Un contexto L-fuzzy se define como una tupla (L, X, Y, R) donde L es un retículo completo, X e Y son respectivamente el conjunto de objetos y atributos, y la relación $R \in L^{X\times Y}$ toma valores en el retículos $(L, \leq)$.

Para extraer información de estos contextos L-fuzzy, el operador derivación se define para todos los conjuntos $A \in L^X$ y $B \in L^Y$ mediante las siguientes expresiones [9]:

$$A_1(y) = \inf_{x\in X}\{I(A(x), R(x, y))\}, \forall y \in Y,$$
$$B_2(x) = \inf_{y\in Y}\{I(B(y), R(x, y))\}, \forall x \in X.$$

Siendo I un operador implicación difuso definido en L.

Los conceptos L-fuzzy constituyen una herramienta que permite visualizar la información almacenada en el contexto. Estos conceptos son pares (M, M_1) donde el conjunto $M \in L^X$ es un punto fijo del operador constructor φ, que se define usando los operadores de derivación $\varphi(A) = (A_1)_2 = A_{12}$ para todo $A \in L^X$.

De manera equivalente, los conceptos L-fuzzy se pueden definir desde el punto de vista de los atributos como pares (N_2, N) siendo $N \in L^Y$ un punto fijo del operador constructor ϕ que se define como $\phi(B) = (B_2)_1 = B_{21}$ para todo $B \in L^Y$.

Dado un subconjunto de objetos $A \in L^X$ (o, un subconjunto de atributos $B \in L^Y$), es posible calcular el concepto L-fuzzy asociado aplicando repetidamente el operador constructor φ (o, equivalentemente, el operador ϕ) hasta que un punto fijo es obtenido. Podemos obtener un punto fijo simplemente aplicando el operador derivación dos veces si el operador de implicación usado es residuado [12], [13]. Así, si I es residuado, entonces el par (A_{12}, A_1) es el concepto L-fuzzy asociado a A. De manera similar, el concepto L-fuzzy asociado a $B \in L^Y$ será el par (B_2, B_{21}).

El concepto L-fuzzy asociado a un subconjunto de objetos A (o atributos B) proporciona la situación "estable" en el contexto que sea más cercana a las condiciones iniciales fijadas por A (o B). En este trabajo, y con el fin de simplificar los cálculos, el operador de implicación elegido para obtener los conjuntos derivados fue la implicación de Lukasiewicz.

III-B. Metodología propuesta para extraer información de los datos experimentales

La metodología propuesta en este trabajo aprovecha el potencial que tiene el análisis de contextos L-fuzzy para extraer información de tablas de relaciones con doble entrada, y se puede describir en los siguientes pasos:

1. Determinar los conjuntos de objetos y atributos que forman parte del problema. La relación entre ellos vendrá dada por los datos experimentales.
2. Normalizando los valores experimentales, construir el contexto L-fuzzy relacionado con las propiedades del material a analizar.
3. Establecer el conjunto de atributos que representa la situación que queremos estudiar. Para las propiedades para las que no tenemos requerimientos iniciales se tomará el valor de 0.
4. Calcular el concepto L-fuzzy más cercano a la situación de partida, que vendrá dada por el punto fijo del operador constructor.
5. Interpretar los valores obtenidos para extraer la información requerida.

El Algoritmo 1 describe los cálculos a realizar para obtener los valores estimados de las propiedades de los materiales.

IV. DETERMINACIÓN DE LA FORMULACIÓN DEL FILM BASADO EN GELATINA

IV-A. Diseño experimental para la optimización de la formulación

Los valores de entrada de la formulación a ajustar elegidos fueron el contenido de glicerol, pH y contenido de ácido gálico. Los datos para poder aplicar la metodología difusa propuesta, fueron distintas muestras donde los valores de contenido de glicerol analizados fueron 0 %, 5 % y 10 %, los valores de pH seleccionados 4.5, 7.25 y 10, mientras que los valores de contenido de ácido gálico fueron 5 %, 10 % y 15 %. Los valores experimentales de las propiedades de salida obtenidas para alimentar los cálculos se muestran en la Tabla II.

IV-B. Construcción del contexto L-fuzzy que representa los valores experimentales

Con el objetivo de modelar el proceso de diseño de materiales construimos un contexto L-fuzzy (L, X, Y, R) en el retículo $(L = [0,1]_{100}, \leq)$, considerando el conjunto de objetos $X = \{x_1, x_2, \ldots, x_{13}\}$, donde x_i representa la i-ésima formulación. El conjunto de atributos $Y = \{y_1, y_2, \ldots, y_{10}\}$ estuvo formado por las diferentes propiedades que se analizaron en cada formulación:

- y_1: Contenido en glicerol
- y_2: Contenido en ácido gálico
- y_3: pH de la solución
- y_4: Ángulo de contacto de agua
- y_5: Tasa de transmisión de vapor de agua
- y_6: Valor de color L*
- y_7: Valor de color b*

Algoritmo 1 Estimación de las propiedades de los materiales

Entradas:

1: $\{x_1, x_2, \ldots, x_n\}$: conjunto de objetos X.
2: $\{y_1, y_2, \ldots, y_m\}$: conjunto de atributos Y.
3: $E(x_i, y_j)$ para todo $(x_i, y_j) \in X \times Y$: valores experimentales.
4: $\{r_1, r_2, \ldots, r_k\}$: requerimientos para los atributos $\{y_{j_1}, y_{j_2}, \ldots, y_{j_k}\}$.

Salida:

$\{e_1, e_2, \ldots, e_m\}$: valores estimados para los atributos.

Pasos:

► Construcción de la relación del contexto.

1: **for** $i = 1$ **to** n **do**
2: **for** $j = 1$ **to** m **do**
3: $$R(x_i, y_j) \leftarrow \frac{E(x_i, y_j) - \min\limits_{1 \leq h \leq n} \{E(x_h, y_j)\}}{\max\limits_{1 \leq h \leq n} \{E(x_h, y_j)\} - \min\limits_{1 \leq h \leq n} \{E(x_h, y_j)\}}$$
4: **end for**
5: **end for**

► Construcción del conjunto inicial de atributos para representar la información requerida.

6: **for** $j = 1$ **to** m **do**
7: **if** $j \in \{j_1, j_2, \ldots, j_k\}$ **then**
8: $$B(y_j) \leftarrow \frac{r_j - \min\limits_{1 \leq h \leq n} \{E(x_h, y_j)\}}{\max\limits_{1 \leq h \leq n} \{E(x_h, y_j)\} - \min\limits_{1 \leq h \leq n} \{E(x_h, y_j)\}}$$
9: **else**
10: $B(y_j) \leftarrow 0$
11: **end if**
12: **end for**

► Cálculo del concepto asociado.

13: **while** $B \neq \phi(B)$ **do**
14: $B \leftarrow \phi(B)$
15: **end while**

► Obtención de los valores estimados para las propiedades.

16: **for** $j = 1$ **to** m **do**
17: $$e_j \leftarrow \min_{1 \leq h \leq n} \{E(x_h, y_j)\} + B(y_j) \cdot \left(\max_{1 \leq h \leq n} \{E(x_h, y_j)\} - \min_{1 \leq h \leq n} \{E(x_h, y_j)\} \right)$$
18: **end for**

- y_8: Tensión a tracción.
- y_9: Elongación a rotura
- y_{10}: Valor de brillo a un ángulo de incidencia de 60°

La relación entre objetos y atributos $R \in L^{X \times Y}$ se calculó normalizando los elementos de la matriz E formada por los valores experimentales promedio que se muestran en la Tabla II de la siguiente manera:

$$R(x_i, y_j) = \frac{E(i,j) - \min\limits_{1 \leq h \leq 13} \{E(h,j)\}}{\max\limits_{1 \leq h \leq 13} \{E(h,j)\} - \min\limits_{1 \leq h \leq 13} \{E(h,j)\}}$$

La relación obtenida se muestra en la Tabla III.

Tabla II
VALORES EXPERIMENTALES DE LAS PROPIEDADES FUNCIONALES PARA DIFERENTES FORMULACIONES DE MATERIALES DE ORIGEN RENOVABLE

Formulación			Propiedades de los materiales						
GLY (%)	GA (%)	pH	CA (°)	WVTR $\left(\frac{g}{m^2 \cdot día}\right)$	L*	b*	TS (MPa)	EB (%)	$\text{Brillo}_{60°}$ (Unidades de brillo)
0	10	4.50	74.8 ± 4.2	1057.7 ± 11.5	95.3 ± 0.5	5.4 ± 0.4	78.7 ±12.5	3.8 ± 1.5	125.4 ± 3.2
5	15	4.50	74.6 ± 7.8	1147.3 ± 16.5	95.5 ± 0.2	5.4 ± 0.8	83.4 ± 8.4	2.7 ± 0.6	155.4 ± 10.3
5	10	7.25	52.8 ± 2.3	1272.1 ± 13.4	50.2 ± 6.6	44.6 ± 6.9	79.8 ± 3.7	3.1 ± 0.4	101.6 ± 8.5
5	5	4.50	58.9 ± 2.2	1097.3 ± 8.0	95.7 ± 0.4	4.8 ± 0.5	80.5 ± 2.1	3.9 ± 0.4	148.0 ± 9.9
0	10	10.00	38.3 ± 2.5	1256.3 ± 10.2	32.5 ± 1.3	20.7 ± 1.8	56.3 ± 5.4	2.7± 0.4	50.1 ± 5.6
10	10	10.00	44.6 ± 1.9	1198.0 ± 22.3	26.6 ± 0.5	8.8 ± 0.5	57.9 ± 1.2	2.6 ± 0.2	29.6 ± 3.7
5	5	10.00	44.7 ± 4.6	1227.0 ± 10.6	69.6 ± 5.7	59.1 ± 3.2	70.8 ± 5.6	3.1 ± 0.4	23.5 ± 2.6
0	5	7.25	56.7 ± 1.8	1206.0 ± 8.5	47.2 ± 0.7	38.6 ± 0.7	77.9 ± 10.4	3.5 ± 0.5	28.6 ± 2.5
10	15	7.25	99.3 ± 2.5	1310.3 ± 19.4	46.6 ± 1.8	42.8 ± 1.5	72.1 ± 6.2	2.7 ± 0.4	100.0 ± 6.4
5	15	10.00	40.8 ± 2.5	1001.7 ± 15.3	23.8 ± 0.9	4.0 ± 0.8	62.3 ± 4.0	2.6 ± 0.3	70.9 ± 3.2
10	10	4.50	101.0 ± 3.7	960.3 ± 22.5	95.2 ± 0.2	6.5 ± 0.9	81.8 ± 8.0	3.1 ± 0.6	148.4 ± 6.1
10	5	7.25	68.2 ± 4.7	1220.0 ± 20.7	52.8 ± 3.9	48.6 ± 1.3	77.5 ± 4.1	3.6 ± 0.4	97.6 ± 1.6
0	15	7.25	67.7 ± 4.2	1304.3 ± 9.3	54.1 ± 1.0	49.8 ± 1.5	86.4 ± 7.9	3.2 ± 0.6	100.0 ± 7.0

Tabla III
RELACIÓN DEL CONTEXTO L-FUZZY

R	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	y_9	y_{10}
x_1	0.00	0.50	0.00	0.58	0.28	0.99	0.03	0.66	0.52	0.77
x_2	0.50	1.00	0.00	0.58	0.53	1.00	0.02	0.80	0.02	1.00
x_3	0.50	0.50	0.50	0.23	0.89	0.37	0.74	0.70	0.21	0.59
x_4	0.50	0.00	0.00	0.33	0.39	1.00	0.01	0.72	0.56	0.94
x_5	0.00	0.50	1.00	0.00	0.85	0.12	0.30	0.00	0.02	0.20
x_6	1.00	0.50	1.00	0.10	0.68	0.04	0.09	0.04	0.00	0.05
x_7	0.50	0.00	1.00	0.10	0.76	0.64	1.00	0.43	0.21	0.00
x_8	0.00	0.00	0.50	0.29	0.70	0.32	0.63	0.64	0.35	0.04
x_9	1.00	1.00	0.50	0.97	1.00	0.32	0.70	0.47	0.02	0.58
x_{10}	0.50	1.00	1.00	0.04	0.12	0.00	0.00	0.18	0.00	0.36
x_{11}	1.00	0.50	0.00	1.00	0.00	0.99	0.04	0.76	0.19	0.95
x_{12}	1.00	0.00	0.50	0.48	0.74	0.40	0.81	0.63	0.40	0.56
x_{13}	0.00	1.00	0.50	0.47	0.98	0.42	0.83	0.89	0.24	0.58

IV-C. Conceptos L-fuzzy asociados a los requerimientos iniciales

Una vez definido el contexto L-fuzzy (L, X, Y, R), se estimaron las características del material a partir de los conceptos L-fuzzy asociados con el conjunto de atributos que representan los requisitos deseados que se muestran en la Tabla I.

Los valores requeridos para las propiedades se normalizaron para obtener valores en L, y se supuso inicialmente que los valores de pertenencia de los atributos que representan las propiedades desconocidas eran 0.

Por lo tanto, las propiedades del material para la aplicación de envases de alimentos grasos fueron representadas por el siguiente conjunto difuso de atributos:

$$B = \{y_1/0, y_2/0, y_3/0, y_4/0.82, y_5/0, y_6/0.99, y_7/0.65, y_8/0.11, y_9/1, y_{10}/0.20\}$$

Y, después calcular el concepto L-fuzzy asociado, nos centramos en su intensión:

$$B_{21} = \{y_1/0.63, y_2/0.60, y_3/0.63, y_4/0.89, y_5/0.81, y_6/0.99, y_7/0.65, y_8/0.98, y_9/1, y_{10}/0.70\}$$

La intensión de un concepto se puede interpretar como un conjunto de atributos que siempre se encuentran juntos en el contexto. Por lo tanto, el conjunto de atributos B_{21} obtenido se puede interpretar como el conjunto de atributos más cercano al conjunto requerido B que se dan al mismo tiempo.

A partir de los valores de pertenencia de la intensión de los conceptos L-fuzzy obtenidos, invirtiendo el proceso de normalización, se estimó que, para el envasado de alimentos grasos, las propiedades del material deberían ser las que se muestran en la Tabla IV.

V. RESULTADOS Y DISCUSIÓN

Teniendo en cuenta los datos experimentales para el análisis de contexto L-fuzzy, las formulaciones estimadas para un rendimiento óptimo son las dadas en la Tabla IV. Para la aplicación de envasado de alimento graso, la metodología sugiere valores alrededor del 6.3 % de contenido en glicerol, 11 % de contenido en ácido gálico, y un pH de 8.

Con estos valores se realizaron medidas experimentales de las propiedades funcionales en el laboratorio, obteniéndose los valores presentados en la Tabla V.

Como puede observarse en la Tabla V, los valores estimados fueron muy similares a los valores objetivo de las propiedades funcionales, excepto para los valores de brillo y tensión a tracción, y los valores propuestos por el análisis de conceptos L-fuzzy fueron mayores que los valores objetivo seleccionados. En lo que respecta a valores de brillo, no es posible obtener valores cercanos a 50 teniendo en cuenta las restricciones

Tabla IV
FORMULACIÓN Y PROPIEDADES ESTIMADAS PARA LOS MATERIALES DE ENVASADO DE ALIMENTOS GRASOS

GLY (%)	GA (%)	pH	CA (°)	WVTR $\left(\frac{g}{m^2 \cdot día}\right)$	L*	b*	TS (MPa)	EB (%)	Brillo$_{60°}$ (Unidades de brillo)
6.28	10.96	7.95	93.94	1243.66	95.00	40.00	89.38	5.00	116.27

Tabla V
VALORES REQUERIDOS PARA EL MATERIAL, ESTIMACIÓN, VALORES EXPERIMENTALES Y ERROR OBTENIDO.

	Formulación			Propiedades de los materiales						
	GLY (%)	GA (%)	pH	CA (°)	WVTR $\left(\frac{g}{m^2 \cdot día}\right)$	L*	b*	TS (MPa)	EB (%)	Brillo$_{60°}$ (Unidades de brillo)
Valores requeridos				90		95	40	60	5	50
Valores estimados	6.3	11	8	93.9	1243.7	95	40	89.4	5	116.3
Valores experimentales				74.3	1370.3	60.9	42.1	83.3	4.8	95.3
Error(%)				20.9	10.2	35.9	5.3	6.7	4.9	18

en el resto de propiedades; esto indica que los valores de brillo deben ser mayores. Como los valores de brillo están relacionados inversamente con la rugosidad de la superficie del film [14], valores mayores de brillo indican superficies más lisas. Algo similar sucedió con los valores de tensión a tracción: los valores estimados fueron mayores que el objetivo. Es interesante señalar que el comportamiento mecánico del material propuesto por el método de estimación es incluso mejor que el de la especificación, pues tiene mayor tensión a tracción, sin disminuir los valores de la elongación a rotura. Para envasado de alimentos grasos, la tasa de transmisión de vapor de agua no es una propiedad crucial, y es por ello que no se seleccionó ningún valor objetivo; pero el concepto L-fuzzy estima un valor de 1243 $\frac{g}{m^2 \cdot día}$ para esta propiedad. Para verificar el método de estimación empleado, se desarrolló un material con la formulación propuesta, y se midieron las propiedades funcionales respuesta del mismo. Como se muestra en la Tabla V, para la mayoría de propiedades, los valores propuestos por el contexto L-fuzzy son razonablemente similares a los valores experimentales. Los errores obtenidos son menores al 21 % en todos los casos, excepto para el valor de color L*, para el que el error es mayor y el valor experimental obtenido es menor que la estimación. Esto produce una luminosidad mayor para el film, mayor que la requerida en la especificación, lo cual es adecuado para la aplicación de envasado de alimentos. Además, se puede confirmar, tal y como predijo el contexto L-fuzzy, que los valores de brillo fueron mayores que los valores objetivo y el comportamiento mecánico del material es incluso mejor que el requerido para la formulación estimada.

VI. CONCLUSIONES

En el presente trabajo, se empleó el análisis de conceptos L-fuzzy como una herramienta de toma de decisiones para obtener una formulación del material con las propiedades funcionales requeridas en envasado de alimentos grasos. Los resultados obtenidos después de implementar la metodología propuesta revelaron que los valores predichos por los conceptos L-fuzzy fueron muy similares a los valores objetivo en la mayoría de propiedades. Además, después de analizar el comportamiento de las formulaciones estimadas, los resultados experimentales resultaron ser muy cercanos a los valores aproximados propuestos. El error entre valores estimados por los conceptos L-fuzzy y los valores experimentales de las formulaciones propuestas fue inferior al 20 % en la mayoría de propiedades funcionales. Por otro lado, en los casos en que la diferencia entre valor predicho y valor experimental fue mayor del 20 %, la predicción proporciona una mejora de las propiedades requeridas, obteniendo un mejor comportamiento del material.

Definitivamente, las formulaciones propuestas por el análisis de conceptos L-fuzzy son el mejor punto de partida para estudiar el efecto de cada parámetro en las propiedades funcionales del material. En vez de tener una enorme matriz de formulaciones que estudiar, utilizar este método de estimación ayuda a identificar aproximadamente los valores de los parámetros de la formulación que llevan a obtener las propiedades deseadas, y permite saber si es posible conseguir las propiedades funcionales para un sistema específico.

AGRADECIMIENTOS

Los autores agradecen a los grupos de investigación del Gobierno Vasco (IT1256-19) y de la Universidad del País Vasco UPV/EHU (GIU18/154).

REFERENCIAS

[1] A. Balakrishna, G. R. Chandra, B. Gogulamudi, C. Someswararao: Fuzzy Approach to the Selection of Material Data in Concurrent Engineering Environment, *Engineering* 3 (9), pp. 921–927, 2011.

[2] T. W. Liao: A fuzzy multicriteria decision-making method for material selection, *Journal of Manufacturing Systems* 15 (1), pp. 1–12, 1996.

[3] H. Haghighi, S. K. Leugoue, F. Pfeifer, H. W. Siesler, F. Licciardello, P. Fava, A. Pulvirenti: Development of antimicrobial films based on chitosan-polyvinyl alcohol blend enriched with ethyl lauroyl arginate (LAE) for food packaging applications, *Food Hydrocolloids* 100 , pp. 105419, 2020.

[4] V. K. Pandey, S. N. Upadhyay, K. Niranjan, P. K. Mishra: Antimicrobial biodegradable chitosan-based composite Nano-layers for food packaging, *International Journal of Biological Macromolecules* 157 , pp. 212–219, 2020.

[5] R. Priyadarshi, J. W. Rhim: Chitosan-based biodegradable functional films for food packaging applications, *Innovative Food Science & Emerging Technologies* 62 , pp. 102346, 2020.

[6] I. Leceta, P. Guerrero, K. de la Caba: Functional properties of chitosan-based films, *Carbohydrate Polymers* 93 (1) , pp. 339–346, 2013.
[7] A. Burusco, R. Fuentes-González: The study of the L-fuzzy concept lattice, *Mathware and Soft Computing* 1 (3), pp. 209–218, 1994.
[8] A. Burusco, R. Fuentes-González: Construction of the L-fuzzy Concept Lattice, *Fuzzy Sets and Systems* 97 (1), pp. 109–114, 1998.
[9] A. Burusco, R. Fuentes-González: Concept lattices defined from implication operators, *Fuzzy Sets and Systems* 114 (3), pp. 431–436, 2000.
[10] B. Ganter, R. Wille: Formal concept analysis: Mathematical foundations, Springer, Berlin - New York, 1999.
[11] R. Wille: Restructuring lattice theory: an approach based on hierarchies of concepts, in: *Rival I. (Ed.), Ordered Sets, Reidel, Dordrecht-Boston*, pp. 445-470, 1982.
[12] R. Bêlohlávek: Fuzzy Galois Connections, *Mathematical Logic Quarterly* 45 (4), pp. 497–504, 1999.
[13] S. Pollandt: Fuzzy-Begriffe: Springer Berlin Heidelberg, 1997.
[14] L. Sánchez-González, M. Cháfer, A. Chiralt, C.o González-Martínez: Physical properties of edible chitosan films containing bergamot essential oil and their inhibitory action on Penicillium italicum, *Carbohydrate Polymers* 82 (2), pp. 277–283, 2010.

Una versión ternaria del algoritmo de Quine McCluskey para la minimización de base de reglas difusas

Leonardo Jara
Departamento de Ciencias de la Computación e Inteligencia Artificial, Instituto Interuniversitario de Investigación en Data Science and Computational Intelligence (DaSCI) Universidad de Granada
18071-Granada, España
E-mail: ljara@correo.ugr.es

Antonio González
Departamento de Ciencias de la Computación e Inteligencia Artificial Instituto Interuniversitario de Investigación en Data Science and Computational Intelligence (DaSCI) Universidad de Granada
18071-Granada, España
E-mail: A.Gonzalez@decsai.ugr.es

Raúl Pérez
Departamento de Ciencias de la Computación e Inteligencia Artificial Instituto Interuniversitario de Investigación en Data Science and Computational Intelligence (DaSCI) Universidad de Granada
18071-Granada, España
E-mail: fgr@decsai.ugr.es

***Resumen*—Los sistemas de clasificación basados en reglas difusas permiten en muchos casos obtener un conocimiento fácil de interpretar, pero esto no siempre ocurre. El objetivo de este trabajo consiste en mejorar la interpretabilidad de estos sistemas, utilizando procesos de minimización de conjuntos de reglas difusas basados en una modificación del método de Quine McCluskey, que permite trabajar con valores ternarios. De este modo, se pretende mejorar la capacidad de eliminación de variables irrelevantes en los antecedentes de las reglas y por lo tanto, mejorar la interpretabilidad del modelo sin alterar en gran medida la precisión.**

***Palabras Clave*—Reglas difusas, Clasificación, Interpretabilidad, Simplicidad, Minimización.**

I. INTRODUCCION

Tradicionalmente la gran ventaja de los sistemas de clasificación basados en reglas difusas es que son capaces de obtener un conocimiento interpretable a la vez que consiguen buenos niveles de precisión. Sin embargo, esto no es siempre así, y en algunos casos el conocimiento obtenido no es tan fácil de interpretar. Este punto es importante ya que un algoritmo que obtenga conocimiento interpretable permite al usuario entender el proceso seguido en la toma de decisiones. La interpretabilidad es un tema de discusión frecuente [1] y existe una gran cantidad de estudios y métricas que nos permiten comparar modelos. Uno de los parámetros frecuentemente utilizado para estudiar la interpretabilidad de un sistema es el número de reglas, de forma que cuanto mayor sea menor es la interpretabilidad.

Existen estudios que buscan mejorar la interpretabilidad de la base de reglas como el CFM-BD [2] que utiliza técnicas de probabilidad e inducción de reglas, creando así una base de reglas más compacta. También tenemos los que buscan reducir el número de reglas difusas utilizando los mapas de Karnaugh [3]. El mapa de Karnaugh es un método gráfico que se utiliza para simplificación de funciones algebraicas de forma canónica. El principal inconveniente con esta técnica radica en que solo puede usar un máximo de 6 variables binarias, las cuales solo pueden representar dos variables difusas con 3 etiquetas lingüísticas, lo que limita bastante su uso.

Este problema también lo encontramos en uno de los algoritmos de clasificación de reglas difusas basados en el algoritmo de Wang y Mendel para clasificación [4], conocido como el algoritmo de CHI [5]. Este algoritmo tiene como ventaja el ser muy eficiente en el proceso aprendizaje a la vez que obtiene niveles de precisión aceptables, pero como inconveniente genera un gran número de reglas difusas [6], lo que dificultad la interpretabilidad del modelo obtenido.

Debido a estos inconvenientes se propuso un nuevo método para minimizar la base de reglas difusas [7], basado en el algoritmo de Quine McCluskey [8], en donde se consiguieron buenos resultados, pero sus principales inconvenientes se encontraban en la adaptación de las etiquetas lingüísticas a una codificación binaria, lo que creaba tablas de la verdad muy extensas y no lograba eliminar variables.

Quine McCluskey es un método de minimización de funciones booleanas ampliamente utilizado para la minimización de circuitos lógicos, debido a que obtiene el mínimo valor de una función booleana con un método tabular. La lógica binaria es eficiente y potente, pero es uno de los inconveniente para cierto tipo de problemas como el propuesto en este trabajo donde queremos trabajar con un numero de tres etiquetas lingüísticas.

En este trabajo proponemos el uso de una lógica ternaria [9] que reemplaza los valores de 0 y 1 de la lógica binaria por valores 0, 1 y 2, lo que nos permitirá realizar una reducción más eficiente del conjunto de reglas y poder mejorar por tanto la interpretabilidad del sistema.

Este documento los dividimos de la siguiente forma, en

Trabajo cofinanciado por el Programa Nacional de Becas de Postgrado en el Extranjero "Don Carlos Antonio López, BECAL", otorgado por el Gobierno de la República del Paraguay y por el proyecto de investigación RTI2018-098460-B-I00 financiado a su vez con fondos FEDER (Unión Europea).

la siguiente sección describimos las herramientas preliminares del trabajo, en la sección III se explicará en detalle el funcionamiento del algoritmo. La sección IV muestra los resultados obtenidos con el método de minimización ternario comparándolo con el modelo de minimización binario, y por último en la sección V se realiza un análisis de la experimentación realizada.

II. PRELIMINARES

Describimos en esta sección las herramientas básicas que se requieren en la realización de nuestra propuesta.

II-A. Modelo de regla

La propuesta de minimización que realizamos en este trabajo parte de un conjunto de reglas difusas básicas que podrían obtenerse de múltiples formas, sin embargo, con la idea de poder realizar una experimentación completa del modelo, asumiremos que el conjunto de reglas inicial se ha obtenido a partir del algoritmo Chi.

La estructura habitual de un conjunto de reglas difusas utilizando el algoritmo Chi es [10]:

$$\begin{aligned}\text{Rule } R_j: \text{ If } x_1 \text{ is } A_{j1} \text{ and ... and } x_n \text{ is } A_{jn} \text{ then}\\ \text{Class } = C_j \text{ with } RW_j\end{aligned} \quad (1)$$

Donde R_j es la etiqueta j-ésima de la regla, $x = (x_1, ..., x_n)$ es un vector n dimensional que representa un ejemplo, A_{ji} es conjunto difuso representado por una función de pertenencia triangular, C_j es la etiqueta de la clase y RW_j es el peso de la regla. El peso de la regla utilizado en este trabajo es el factor de certeza penalizado [11] PCF:

$$RW_j = PCF = \frac{\sum_{x_p \in ClassC_j} \mu_{A_j}(x_p) - \sum_{x_p \notin ClassC_j} \mu_{A_j}(x_p)}{\sum_{p=1}^{p} \mu_{A_j}(x_p)} \quad (2)$$

Donde $\mu_{Aj}(x_p)$ es el grado de pertenencia del ejemplo x_p con la parte del antecedente de la regla difusa R_j y se calcula como sigue.

$$\mu_{A_j}(X_p) = \prod_{i=1}^{n} \mu_{A_{ji}}(X_{pi}) \quad (3)$$

En donde $\mu_{A_{ji}}(X_{pi})$ es el grado de pertenencia del valor X_{pi} del conjunto difuso A_{ji} de la regla R_j.

Denominaremos *modelo de reglas difusas extendido* [12] al modelo de regla difusa que permite que el valor asignado a una variable sea un subconjunto de las etiquetas difusas de su dominio. Este modelo de regla es interesante, ya que cuando una variable toma todas las etiquetas de su dominio, la variable se puede considerar irrelevante y puede eliminarse del antecedente de la regla. En la propuesta realizada en este trabajo, las reglas resultantes de la simplificación serán de este tipo.

II-B. Método de Quine McCluskey

En esta sección presentamos las definiciones básicas y los pasos para aplicar el método Quine McCluskey [6].

Definiciones

- Literal: Es una variable lógica o su negación (q or $\bar{q}$).
- Minterm: es una expresión algebraica booleana de n variables booleanas, que solamente se evalúa como verdadera para una única combinación de esas variables.
- Implicante Primo: es el producto que no se puede combinar con otro término para eliminar una variable y una mayor simplificación.
- Implicante Primo esencial: es un implicante primo que es capaz de cubrir una salida de la función que no está cubierta por ninguna combinación de implicantes primos.

El método de Quine McCluskey (QM) utiliza las siguientes tres leyes básicas de simplificación.

- $q + \bar{q} = 1$ (Complemento)
- $q + q = q$ (Idempotente)
- $q(w + z) = qw + qz$ (Distributiva)

Donde *q, w y z* son literales.

El método QM tiene los siguientes pasos:

- Encontrar el implicante primo: En este paso, se sustituye el literal en forma de 0 y 1 y generamos una tabla. Inicialmente el número de filas de la tabla es igual al número total de minterms de la función original no simplificada. Si dos términos sólo se diferencian en un bit, una variable aparece en ambas formas (variable y negación), entonces se podrá utilizar la ley del complemento. Iterativamente, se comparan todos los términos y se genera el implicante primo.
- Encontrar el implicante primo esencial: Usando los implicantes primos del paso anterior, se genera una tabla para encontrar los implicantes primos esenciales. Algunos implicantes primos pueden ser redundantes y pueden ser omitidos, pero si aparecen sólo una vez, no pueden ser omitidos y proporcionan implicantes primos esenciales.
- Encontrar otro implicante primo: No es necesario que el implicante primo esencial cubra todos los términos mínimos. En ese caso, se considerara otros implicantes primos para asegurarnos de que todos los términos menores han sido cubiertos.
- Cuando no podemos obtener ni un implicante primo esencial estamos ante un problema de cobertura cíclica, que puede ser resuelto analíticamente por el método de Petrick's [13].

En general, el método QM proporciona un método mejor para la simplificación de una función booleana que el mapa de Karnaugh, pero sigue siendo un problema NP-Duro, por lo que la complejidad de vuelve exponencial.

III. MÉTODO PROPUESTO PARA LA MINIMIZACIÓN DE REGLAS

En [7] se propone una técnica que toma como base el modelo de minimización de circuitos digitales de Quine McCluskey para la simplificación y reducción de bases de reglas difusas para clasificación. Dicha técnica considera que los antecedentes de todas las reglas que describen una clase se pueden considerar como un único circuito y la idea es aplicar los criterios de reducción que se describen en el modelo de Quine McCluskey. En las conclusiones del trabajo, los autores describen que el uso de la base binaria produce unos resultados de reducción aceptables, pero que el sistema tiene problemas para eliminar variables completas en el antecedente de la regla y eso limita la capacidad de simplificación y por tanto la mejora en la interpretabilidad del modelo resultante.

En este trabajo proponemos un método de minimización de conjuntos de reglas difusas extendiendo el algoritmo Quine McCluskey de minimización de funciones booleanas para trabajar con lógica ternaria. El algoritmo se extiende para trabajar con tablas de verdad ternarias. Al trabajar con lógica ternaria y con conjuntos de reglas difusas que codifican sus variables difusas usando 3 etiquetas, se mejora la capacidad de eliminación de variables completas en el antecedente de la reglas.

El proceso que se sigue es el siguiente: (A) partimos de un conjunto de reglas difusas. En general, entenderemos que las reglas siguen el modelo conocido como ”Mamdani”, como se describe en la ecuación (1). Además, vamos a suponer que el dominio asociado a cada variable difusa está compuesto por 3 etiquetas uniformemente distribuidas sobre el universo de discurso de la variable. (B) Se transforma el antecedente de cada una de las reglas en un vector con codificación ternaria y a cada uno de estos vectores se le asocia la clase de la regla. (C) Se toma una de las posibles clases C_j y todos los vectores transformados asociados a esa clase y se aplica sobre ellos el algoritmo de minimización de Quine McCluskey ternario. (D) El nuevo conjunto de vectores obtenidos, asociados a la clase C_j, son transformados en reglas y sustituyen a las reglas originales para esa clase. (E) El proceso vuelve al paso 3 hasta que el método se aplica a todas las clases.

Cada variable lingüística tiene un dominio difuso asociado y el algoritmo de CHI utiliza 3 etiquetas lingüísticas impares, en este caso usaremos las siguientes, *small, medium y large* que se observan en la Figura 1.

III-A. Codificación de la base de reglas.

Como dijimos anteriormente, para utilizar el método de Quine McCluskey es necesario transformar la base de reglas en un formato compatible con el método de Quine McCluskey, donde se utilizan tablas de la verdad binarias. En este trabajo, adaptamos el método utilizando una codificación ternaria y trabajaremos con conjuntos de reglas que usan variables difusas que tienen asociados dominios difusos con 3 etiquetas.

Teniendo como referencia general las tres etiquetas difusas *small, medium, large*, se usará la codificación que se observa en la Figura 2.

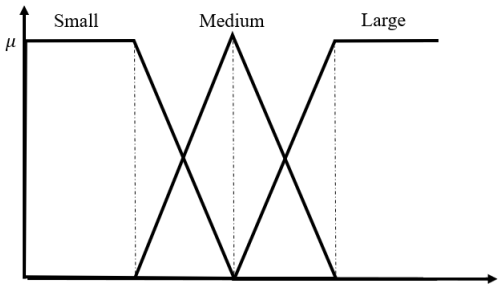


Figura 1. Dominio difuso de una variable lingüística $x_1, ..., x_n$.

Etiqueta	Código
small	0
medium	1
large	2

Figura 2. Codificacion ternaria

III-B. Método de minimización Quine McCluskey ternario

En esta sección describimos la extensión a base ternaria del método de Quine McCluskey. Dicha extensión se puede describir con los siguientes pasos:

1. Convertimos el antecedente de cada regla de la base de reglas a una fila de una tabla de verdad de valores ternarios tomando el sistema de codificación de la Figura 2. Se construye una tabla distinta para cada clase.
2. Sumamos los valores de todas las variables correspondientes a cada fila de esa tabla (cada fila representa el antecedente de cada una de las reglas) y se reordenan las filas en orden ascendente.
3. Creamos una columna con los *minterms* asociados al número de fila a la que pertenece cada regla.
4. Comparamos reglas cuya suma solo difiera en una unidad para buscar elementos redundantes. Si tres reglas difieren en una unidad entre sí y tienen la misma variable con distintas etiquetas difusas, marcaremos esta variable con el símbolo “*” que significa que dicha variable es redundante. Asociamos la nueva regla con los *minterms* pertenecientes a las reglas combinadas.
5. Los términos que no se pueden combinar con tres reglas donde solo una variable tiene las tres etiquetas difusas, se analizan en pares. Comparamos reglas cuya suma difiera en una unidad para buscar elementos redundantes. En caso de que 2 reglas cuya suma difiera en una unidad y tengan una misma variable con distinta etiqueta lingüística, marcaremos esta variable como la unión de las etiquetas difusas. Asociamos la nueva regla con los *minterms* pertenecientes a las reglas combinadas. Los nuevos valores codificados se muestran en la Figura 3.
6. Los términos que no se pueden combinar, los apartamos en otra tabla como implicantes primos con su respectivo *minterm* asociado.

7. Eliminamos los duplicados en la columna del *minterm*.
8. Repetimos el proceso 4,5,6 y 7, agregando las nuevas variables correspondientes a la redundancia "*" del paso 4 y la unión de las etiquetas difusas correspondiente al paso 5, hasta que ninguna fila se puede combinar. Creando así una tabla con los implicantes primos.
9. Para encontrar los implicantes esenciales creamos una tabla de cobertura. Donde las filas corresponden a los *minterm* de las reglas combinadas denominadas implicantes primos del paso 8 y las columnas corresponden a los valores individuales de cada *minterm*. Los minterms correspondientes a las reglas redundantes son omitidos en este paso, no se colocan en las columnas. En la tabla de cobertura los minterms cubren los implicantes primos. Un implicante primo esencial es aquel implicante primo que cubre un minterm y ningún otro implicante primo cubre el mismo minterm.
10. Si los implicantes primos no esenciales no tienen un minterm en común con los implicantes primos esenciales, se vuelve a crear la tabla de cobertura del paso 9 de modo a extraer los nuevos implicantes primos esenciales, Esto se repite hasta no obtener más implicantes primos esenciales.
11. En el caso de que no exista ningún implicante primo esencial se aplica el método de Petrick's.

Etiqueta	**Código**
Small	0
Médium	1
Large	2
Small + Medium + Large	*
Small + Medium	01
Medium + Large	12
Small + Large	02

Figura 3. Codificacion ternaria ampliada.

Veamos un ejemplo para aclarar como funciona el método. Supongamos el siguiente conjunto de reglas difusas, todas de la clase C_0,

R_1: If x_1 is SMALL and x_2 is SMALL and x_3 is MEDIUM and x_4 is LARGE then Class = C_0
R_2: If x_1 is SMALL and x_2 is MEDIUM and x_3 is MEDIUM and x_4 is LARGE then Class = C_0
R_3: If x_1 is SMALL and x_2 is LARGE and x_3 is MEDIUM and x_4 is LARGE then Class = C_0
R_4: If x_1 is SMALL and x_2 is SMALL and x_3 is SMALL and x_4 is LARGE then Class = C_0
R_5: If x_1 is MEDIUM and x_2 is SMALL and x_3 is SMALL and x_4 is LARGE then Class = C_0
R_6: If x_1 is MEDIUM and x_2 is SMALL and x_3 is SMALL and x_4 is MEDIUM then Class = C_0

Convertimos la base de reglas en una tabla de codificación ternaria.

Regla	Minterm	Variable 1	Variable 2	Variable 3	Variable 4	Suma	Clase
R_4	1	0	0	0	2	2	0
R_1	2	0	0	1	2	3	0
R_5	3	1	0	0	2	3	0
R_6	4	1	0	0	1	3	0
R_2	5	0	1	1	2	4	0
R_3	6	0	2	1	2	5	0

Podemos apreciar que las reglas correspondientes a los minterm 2, 5 y 6 tienen todas las etiquetas difusas posibles en la variable 2, por lo que podemos simplificar las reglas 2,5 y 6. Al igual que las reglas 1 y 3 que tienen 2 etiquetas difusas en la variable 1.

Minterm	Variable 1	Variable 2	Variable 3	Variable 4	Clase
2,5,6	0	*	1	2	0
1,3	0-1	0	0	2	0
4	2	0	0	1	0

En la tabla ya no se pueden realizar agrupaciones, a esta tabla la denominamos implicantes primos. Completamos la tabla de cobertura con la idea de encontrar los implicantes primos esenciales.

Minterm	1	2	3	4	5	6
2,5,6		x			x	x
1,3	x		x			
4				x		

Podemos apreciar que cada implicante primo tiene un único valor en alguna columna correspondiente a los minterms. Por esto todos los implicantes primos son esenciales y la base de reglas simplificada queda de la siguiente forma.

IF x_1 is SMALL and x_3 is MEDIUM and x_4 is LARGE then Class = C_0
IF x_1 is SMALL or MEDIUM and x_2 is SMALL and x_3 is SMALL and x_4 is MEDIUM then Class = C_0
IF x_1 is LARGE and x_2 is SMALL and x_3 is SMALL and x_4 is MEDIUM then Class = C_0

donde en la primera regla eliminamos la variable x_2 y en la segunda regla usamos el modelo de reglas difusas extendidas en la variable x_1.

IV. ESTUDIO EXPERIMENTAL

En esta sección queremos comprobar si el método propuesto permite reducir el número de reglas difusas, manteniendo el nivel de precisión, con respecto a los resultados obtenidos por el algoritmo de CHI original, y a la vez si permite obtener reglas más simples que las obtenidas con la propuesta descrita en [7]. Para ello utilizamos las bases de datos descritas en la Tabla I obtenidas del repositorio de la UCI [14]. Usaremos el algoritmo de CHI [5] para generar los conjuntos de reglas.

Serán esos conjuntos de reglas los que usaremos para evaluar el comportamiento de la propuesta y serán comparados con los obtenidos por el método usando codificación binaria [7].

La implementación del código se ha hecho en Python y se han utilizamos dominios difusos compuestos por 3 etiquetas difusas con función de pertenencia triangular simétrica.

El estudio se centra en la comparación de tres parámetros: capacidad de predicción *P*, número de reglas *#R* y la simplicidad de cada base de conocimiento *Simp* en el caso binario *BIN* y en el caso ternario *TER*. En este trabajo se ha considerado la simplicidad como la reducción en las condiciones en los antecedentes de las reglas. Los resultados que se muestran son los obtenidos tras realizar una validación cruzada de 10.

Tabla I
BASES DE DATOS UTILIZADAS EN EL ESTUDIO

Base de datos	***Ejemplos***	***Atributos***	***Número de Clases***
Shuttle	57999	9	9
abalone	3756	8	28
iris	150	4	3
newthyroid	215	5	3
banana	5300	2	2
heart	270	13	2
saheart	462	9	2
bupa	345	6	2
led7digit	500	7	10
page-blocks	5472	10	5
mammographic	830	5	2
flare	1066	11	6
pima	768	8	2
phoneme	5404	5	2
australian	690	14	2
crx	653	15	2
balance	625	4	3
german	1000	20	2
thyroid	7200	21	3
magic	19020	10	2
wine	178	13	13
cleveland	297	13	5
automobile	159	25	6
bands	365	19	2
ionosphere	351	33	2
spectfheart	267	44	2
sonar	208	60	2
movement-libras	360	90	15
penbased	10992	16	10
vehicle	846	18	4
vowel	990	11	11
segment	2310	19	7
winequality-white	4898	11	11
spambase	4597	57	2
ring	7400	20	2

La Tabla II muestra una comparativa de las bases de reglas obtenidas por el algoritmo de aprendizaje (CHI) y los dos métodos de simplificación estudiados, la minimización de la base de reglas utilizando el método de Quine McCluskey Binario (CHI + BIN) y la minimización de la base de reglas utilizando el método de Quine McCluskey Ternario (CHI + TER). Donde *P* representa la precisión y *R* la cantidad de reglas generadas por el algoritmo.

En esta tabla podemos observar que ambos métodos proporcionan una reducción importante en el número de reglas final (quedándose aproximadamente en un 58 % del tamaño del conjunto de reglas original). Comparando los métodos binario y ternario en su capacidad de reducción de reglas, se puede observar que no aparecen diferencias significativas entre ambos métodos. Ambos métodos reducen en valores muy parecidos los conjuntos de reglas como se puede observar por los valores medios obtenidos sobre este parámetro. En cuanto a la capacidad de predicción, podemos observar en la Tabla II que los valores medios de *P* en CHI con respecto a (CHI + BIN) y (CHI + TER), presentan una mínima variabilidad.

Tabla II
RESULTADOS OBTENIDOS

	CHI		**CHI + BIN**		**CHI + TER**	
Base de Datos	***P***	***#R***	***P***	***#R***	***P***	***#R***
Shuttle	0,801	28,7	0,802	17,5	0,802	16,7
abalone	0,231	69,4	0,230	42,5	0,228	41,5
iris	0,926	14,7	0,940	7,9	0,940	7,9
newthyroid	0,846	20,4	0,810	7,9	0,833	9,5
banana	0,602	7,9	0,580	4	0,558	4
heart	0,515	217,2	0,493	165,3	0,504	156,9
saheart	0,727	168,7	0,721	78,5	0,721	74,6
bupa	0,578	43,3	0,573	21,8	0,570	20,9
led7digit	0,670	93,9	0,618	92,9	0,676	50,2
page-blocks	0,918	49,7	0,918	26,7	0,916	24,5
mammographic	0,808	45,5	0,812	19	0,816	17
flare	0,567	127,6	0,580	81	0,492	62,2
pima	0,725	101	0,695	43,9	0,697	41,8
phoneme	0,718	50,1	0,712	17,5	0,709	17,5
australian	0,798	313,9	0,783	179	0,783	139,6
crx	0,813	258,2	0,808	154,6	0,805	113,8
balance	0,893	82	0,840	23,6	0,851	24,1
german	0,196	885,2	0,196	836,1	0,196	824,8
thyroid	0,920	463,6	0,910	268,8	0,919	190,8
magic	0,764	313,1	0,744	103,7	0,744	95,4
wine	0,926	124,3	0,921	86	0,926	81,8
cleveland	0,380	239,2	0,354	194,2	0,374	186,7
automobile	0,611	109,7	0,602	86,4	0,577	81,4
bands	0,686	258,2	0,683	194	0,680	197,9
ionosphere	0,652	228,8	0,655	215,6	0,655	213,3
spectfheart	0,663	235,2	0,663	235,2	0,663	235,2
sonar	0,595	187,2	0,596	185,6	0,596	185,6
movement libras	0,733	260,6	0,731	257,3	0,731	257,3
penbased	0,975	3281	0,973	1352,7	0,972	1427,5
vehicle	0,630	378,6	0,617	211,8	0,618	218,4
vowel	0,639	285,9	0,628	193,9	0,622	189,3
segment	0,843	318,5	0,849	139,3	0,852	136,8
winequality-white	0,495	294	0,494	133,4	0,481	132
spambase	0,728	357	0,693	210,2	0,688	211,1
ring	0,552	573	0,545	428,1	0,536	423,9
Media	0,689	299,580	0,679	180,454	0,678	174,626

En la Tabla III podemos observar una comparación de los métodos de minimización binario y ternario, donde la columna *eliminadas* corresponde a la cantidad media de variables irrelevantes que fueron detectadas por el algoritmo de minimización ternario y las columnas *Simp BIN* y *Simp TER* muestran la medida de simplicidad de cada base de datos para los métodos binario y ternario respectivamente de acuerdo a la ecuación (4).

$$S = \frac{(variables.reglas_{(BIN,TER)}) - eliminadas}{variables.reglas_{(CHI)}} \quad (4)$$

Tabla III
ESTUDIO DE INTERPRETABILIDAD

Base de datos	*Eliminadas*	*Simp BIN*	*Simp TER*
Shuttle	1,0	0,610	0,578
abalone	0,9	0,612	0,596
iris	0,0	0,537	0,537
newthyroid	1,8	0,387	0,448
banana	1,0	0,506	0,443
heart	0,7	0,761	0,722
saheart	7,1	0,465	0,438
bupa	2,3	0,503	0,474
led7digit	0,0	0,989	0,535
page-blocks	3,1	0,537	0,487
mammographic	6,5	0,418	0,345
flare	8,0	0,635	0,482
pima	8,7	0,435	0,403
phoneme	7,6	0,349	0,319
australian	8,4	0,570	0,443
crx	10,3	0,599	0,438
balance	25,0	0,288	0,218
german	0,7	0,945	0,932
thyroid	13,1	0,580	0,410
magic	16,1	0,331	0,300
wine	1,6	0,692	0,657
cleveland	0,0	0,812	0,781
automobile	0,1	0,788	0,742
bands	2,8	0,751	0,766
ionosphere	1,6	0,942	0,932
spectfheart	0,0	1,000	1,000
sonar	0,0	0,991	0,991
movement libras	0,0	0,987	0,987
penbased	29,2	0,412	0,435
vehicle	6,4	0,559	0,576
vowel	0,5	0,678	0,662
segment	10,6	0,437	0,428
winequality-white	14,2	0,454	0,445
spambase	19,6	0,589	0,590
ring	23,4	0,747	0,738
Media	6,6	0,626	0,579

En esta última tabla se observa que el método ternario efectivamente es capaz de detectar variables irrelevantes en los antecedentes de las reglas, aunque esa detección no es homogénea. En bases de datos como *iris* o *cleveland* o *sonar* no es capaz de encontrar ninguna variable irrelevante. La razón de esto puede deberse a que el método prioriza la reducción en el número de reglas que la eliminación de variables a la hora de realizar las agrupaciones. Por otro lado, podemos observar que la simplicidad medida como número medio de condiciones, es mejor la aportada por el método ternario. Se han comparado ambos modelos usando el test de Wilcoxon para probar que le simplicidad del modelo (CHI + TER) es mayor que (CHI + BIN), obteniéndose un p-value de 0.9999 y por tanto aceptándose la hipótesis. Este resultado pone de manifiesto que el método ternario permite detectar con más facilidad variables irrelevantes.

V. CONCLUSIONES

En este trabajo se ha presentado una extensión de una técnica para la minimización de conjuntos de reglas difusas basado en la técnica de simplificación de circuitos digitales de Quine McCluskey. Tomando como base una versión previa que trabaja con circuitos binarios. Dicha versión ofrece un buen resultado en cuanto a la reducción del número de reglas, pero no permite, de forma fácil, la detección de variables irrelevantes en los antecedentes de las reglas. Este problema se debe al uso de la codificación binaria. Cambiado a codificación ternaria y tomando los dominios con tres etiquetas lingüísticas, la técnica no sólo reduce las reglas, sino que adquiere mayor capacidad para reducir las condiciones en el antecedente de la regla.

Se ha realizado un estudio experimental, usando el algoritmo de CHI como método para extraer las reglas y comparando la versión binaria con la ternaria que se ha propuesto en este trabajo. Los resultados muestran que la nueva propuesta reduce el número de reglas a niveles semejantes a como lo hacía la binaria. Además, frente a la binaria, las bases de reglas obtenidas usando la minimización ternaria presentan un mayor grado de simplicidad cuando se combina la reducción de las reglas y el número de condiciones en el antecedente de las reglas.

REFERENCIAS

[1] M. J. Gacto, R. Alcalá, y F. Herrera, «Interpretability of linguistic fuzzy rule-based systems: An overview of interpretability measures», Information Sciences, vol. 181, n.º 20, pp. 4340-4360, oct. 2011, doi: 10.1016/j.ins.2011.02.021.

[2] M. Elkano, J. A. Sanz, E. Barrenechea, H. Bustince, y M. Galar, «CFM-BD: A Distributed Rule Induction Algorithm for Building Compact Fuzzy Models in Big Data Classification Problems», IEEE Transactions on Fuzzy Systems, vol. 28, n.º 1, pp. 163-177, ene. 2020, doi: 10.1109/TFUZZ.2019.2900856.

[3] M. Karnaugh, «The map method for synthesis of combinational logic circuits», Transactions of the American Institute of Electrical Engineers, Part I: Communication and Electronics, vol. 72, n.º 5, pp. 593-599, nov. 1953, doi: 10.1109/TCE.1953.6371932.

[4] L.X. Wang y J. M. Mendel, "Generating Fuzzy Rules by Learning from Examples," vol. 22, n.º 6, pp. 14, 1992.

[5] Z. Chi, H. Yan y T. Pham, T, "Fuzzy algorithms with applications to image processing and pattern recognition". World Scientific, 1996.

[6] D. Alvarez Estevez y V. Moret Bonillo, «Revisiting the Wang–Mendel algorithm for fuzzy classification», Expert Systems, vol. 35, n.º 4, p. e12268, 2018, doi: 10.1111/exsy.12268

[7] L. Jara, A. González, y R. Pérez, «A preliminary study to apply the Quine McCluskey algorithm for fuzzy rule base minimization», en 2020 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), jul. 2020, pp. 1-6. doi: 10.1109/FUZZ48607.2020.9177739.

[8] P. W. C. Prasad, A. Beg, y A. K. Singh, «Effect of Quine-McCluskey simplification on Boolean space complexity», en 2009 Innovative Technologies in Intelligent Systems and Industrial Applications, Kuala Lumpur, Malaysia, jul. 2009, pp. 165-170. doi: 10.1109/CITISIA.2009.5224219.

[9] A. Dhande, V. Ingole, y V. Ghiye, Ternary Digital System: Concepts and Applications. 2014

[10] M. Elkano, M. Galar, J. Sanz, y H. Bustince, «CHI-BD: A fuzzy rule-based classification system for Big Data classification problems», Fuzzy Sets and Systems, vol. 348, pp. 75-101, oct. 2018, doi: 10.1016/j.fss.2017.07.003.

[11] H. Ishibuchi y T. Yamamoto, «Rule weight specification in fuzzy rule-based classification systems», IEEE Trans. Fuzzy Syst., vol. 13, n.º 4, pp. 428-435, ago. 2005, doi: 10.1109/TFUZZ.2004.841738.

[12] D. García, A. González, y R. Pérez, «Overview of the SLAVE learning algorithm: A review of its evolution and prospects», International Journal of Computational Intelligence Systems, vol. 7, n.º 6, pp. 1194-1221, nov. 2014, doi: 10.1080/18756891.2014.967008.

[13] S.K. Petrick, "On the Minimization of Boolean Functions," in Proceedings of the International Conference Information Processing, pp. 422-423, 1959.

[14] D. Dua, y C. Graff, UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science, 2019

Prometheus: Harnessing Fuzzy Logic and Natural Language for Human-centric Explainable Artificial Intelligence

Ettore Mariotti, Jose M. Alonso-Moral
Centro Singular de Investigación en Tecnoloxías Intelixentes (CiTIUS)
Universidade de Santiago de Compostela
15782, Santiago de Compostela, SPAIN
{ettore.mariotti, josemaria.alonso.moral}@usc.es

Albert Gatt
Institute of Linguistics and
Language Technology, University of Malta (UM)
Msida, Malta
albert.gatt@um.edu.mt

***Abstract*—Prometheus is an interpretable model which is suitable for the generation of visual and textual explanations grounded in common sense knowledge. This model can be seen as a special case of generalized additive models, which can be also interpreted as a list of (fuzzy) rules. The goodness of the model is illustrated with one benchmark dataset from the medical domain. Reported results are encouraging. They suggest that Prometheus exhibits a good balance between understandability and classification performance in comparison with other well-known models (e.g., linear models, decision trees or fuzzy rule-based classifiers) which are deemed as interpretable.**

***Index Terms*—Explainable Artificial Intelligence, Interpretable Machine Learning, Shapley Values, Generalized Additive Models, Fuzzy Rule-based Classifiers**

I. Introduction

In Greek mythology, Prometheus was a semi-god who stole a spark of fire from the gods and created humanity from clay [1]. Moreover, he is said to have brought fire (intelligence) to humankind. In this paper, we introduce a new Artificial Intelligence (AI) modelling method whose name takes inspiration from this myth, in that it constitutes a step ahead towards the creation of self-explaining AI models in real-world applications.

Given a classification or regression task, the objective of machine learning has been to mathematically formalise algorithms for maximising some accuracy measure for the given data. A wide variety of models have been created, some of them achieving impressive performance on benchmarking datasets. However, there is a class of problems for which having good performance metrics is not enough. In particular, a system that supports decision-making for fault-critical situations, or where someone has to take responsibility for the action, has limited utility if it does not provide any meaningful insight into its reasoning process. For example, if an advanced AI in a hospital suggests a diagnosis and a cure, the well-being of the patient is ultimately the responsibility of the doctor in charge and he/she is the person who in the end will have the final word about the diagnosis and treatment. Thus a doctor is going to use such an AI-based diagnostic system only if its recommendations are transparently explained, allowing the professional to evaluate their soundness and possibly communicate them to the patient.

It is within this frame of responsibility, trust, fairness, accountability and liability that eXplainable AI (XAI) plays a key role [2]. An AI system that can explain its reasoning is more likely to inspire trust in decision-makers. Recently, XAI has received lots of attention, with proposals for different approaches and tools aimed to answer different questions. In this context, if we want to lower the barrier to access to these tools, then the use of the verbal modality and textual generation can sparkle an interactive communication between humans and AI that will help understandability and the transfer of knowledge [3]–[5]. This is because humans tend to vastly favour the modality of language when explaining their decisions and the reason behind them.

In this paper, we introduce Prometheus, an interpretable-by-design AI model that supports both visual explanations and textual descriptions of its reasoning. We have evaluated Prometheus with the Breast Cancer benchmark dataset [6]. Moreover, we have shown how Prometheus works in comparison with alternative methods in an illustrative example. The rest of the manuscript is organized as follows. Section II introduces related work. Section III describes how to build and interpret Prometheus XAI models. Section IV presents our experimental settings and reports achieved results. Section V concludes the paper and points out future work.

II. Related Work

One important feature that distinguishes between different XAI approaches [7] is whether an explanation is generated after the prediction is done (post-hoc), or whether one tries to design an interpretable model in the first place.

A. Interpretable by design models

In the literature we find three broad classes of interpretable models; several variations exist within each class.

- **Rule-based Systems** (RBSs): a list of rules in the form IF X THEN Y. The more rules there are, the harder it becomes to understand the model.

- **Decision Trees** (DTs): a list of rules nested into a tree structure. At each step of the reasoning process, the model is looking at just one "test" at a time and based on the result decides which branch to descend along the tree until it reaches a leaf that corresponds to the prediction.
- **Generalized Additive Models** (GAMs): a class of models that are only concerned with first-order interactions between the features and the target. This allows the study of the effect of each feature one at a time.

Both RBSs and DTs are relatively easy to interpret and describe in natural language [8], [9]. However, as far as we know, it is rare to find linguistic explanations associated with GAMs (see e.g., [10]). GAMs are models of the form $g(y) = \sum_i f_i(x_i)$ where x_i represents the i-th feature of $\mathbf{x}$ and $g(\cdot)$ is the *link function* (e.g., logistic sigmoid for classification tasks). Each f_i is called a *shape function* [11]. GAMs are interpretable due to the additive nature of the modelling, allowing a user to focus on the contribution of each feature without accounting for interaction effects. In other words, they are a class of models for which partial dependency plots and their generalisation, Individual Conditional Expectations [12], can be computed exactly, they let users follow the process of computation in a sequential manner, and they are not subject to variation effects (because interactions are restricted to first-order, by design, but see below for details on GA2Ms which incorporate interaction terms). The main types of GAM are:

- **Linear Models** (LMs): where $f_i(x) = \beta_i$ and β_i is a static coefficient.
- **Scoring Systems**: similar to LMs, but where β coefficients are discrete and small [13].
- **Spline Systems**: each partial f_i is estimated with a spline.
- **GA2M**: introduces additional terms that explicitly deal with second-order interactions $f_i(x_j, x_k)$ which are described by heatmaps [14].
- **Explainable Boosting Machine** (EBM)[1]: combines small trees and also deals with second-order interactions.
- **GAMs with Neural Networks**: each f_i is estimated with a neural network. They were first proposed by [15].

B. Post-hoc explanations

With black-box models, it is possible to some extent to generate post-hoc explanations of their behaviour, using different techniques. For example, LIME [16] makes a linear approximation to the black-box so that it is locally equivalent. On the other hand, SHAP [17] is inspired by the game-theoretic Shapley Values that assign to the feature x_j a contribution $\phi_j(x_j)$ such that

$$f(\mathbf{x}) - \mathbb{E}_{\forall x}[f(\mathbf{x})] = \sum_i \phi_i(x_i)$$

SHAP is in general NP-hard to compute, but for some models (like tree ensembles [18]) can be computed efficiently. Moreover, it is accessible and usable thanks to its open-source implementation.[2]

[1]Available at https://github.com/interpretml/interpret

[2]Available at https://github.com/slundberg/shap

The interpretation of these methods must nonetheless be conducted carefully as they have been shown to be potentially vulnerable to some adversarial attacks [19].

III. Description of the Prometheus XAI Model

The architecture of Prometheus is depicted in Fig. 1. This is another step toward a human-centric approach to knowledge discovery and human-machine interaction. The user can benefit from flexibility in the customisation of the processing of data according to common-sense expert knowledge. Also, he/she can exploit the insights learnt from the model communicated with visualisations and textual explanations.

The first component of Prometheus is a *discretiser* that discretises continuous variables into k-bins (using a k-means or quantiles strategy) transforming the original instance into a wider vector of 0s and 1s (this can be interpreted as a way of building rules IF x_i IS IN BIN j). Each bin has a semantic meaning that can be linguistically refined by an expert, just by tuning the associated strong fuzzy partition. As an example let's imagine the task of recommending a certain movie to someone given that we have a feature that represents their age: the discretiser would produce a tuple of features stating whether age is *low*, *medium* or *high*. This initial partition can be semantically refined later by the user with personalized labels like *young*, *mid-age* or *old*.

The output of the discretiser is then combined linearly by an LM with zero bias (regularised by an L1 penalty as it incentives a sparse representation)[3]. In this way, a partial score is attributed to each partition and the final prediction (in log-odds space) will be the global aggregation (sum) of the fired partial scores. Going back to our example, a *young* person could contribute with a score of +1.2 (thus representing evidence of the class "Do recommend") while an *old* one could contribute with -1.1 (thus being counter-evidence for the class, thus "Do not recommend"). If the *mid-age* contributes with 0, then it can be omitted for the sake of reducing the cognitive load to the user.

The generated model can be seen both as a GAM where each f_i is piece-wise constant, and as a fuzzy rule list aggregated by the sum operator (e.g., "R_i: IF $feature_i$ is $label_3$ THEN $output_1$ with $w = s$"). This means that we can use the visualisations typically used for GAMs but also the linguistic explanations associated with fuzzy rule-based systems. In particular, the interpretation of the model as a rule list can support the generation of simple linguistic explanations. This can be achieved for example by ad-hoc mapping rules to templates of the form "Instance i has a high value of feature f and this contributes s to the total log-odd score of class C." Naturally, the language generated could also be tailored to the user, or made more or less precise depending on the context. Thus, we can produce a textual description of the process that leads to the final prediction.

We adopt a relatively simple generation strategy for textual descriptions in the present work, noting that more sophisticated

[3]The number of bins and the coefficient for regularisation are automatically chosen with cross-validation on the training data.

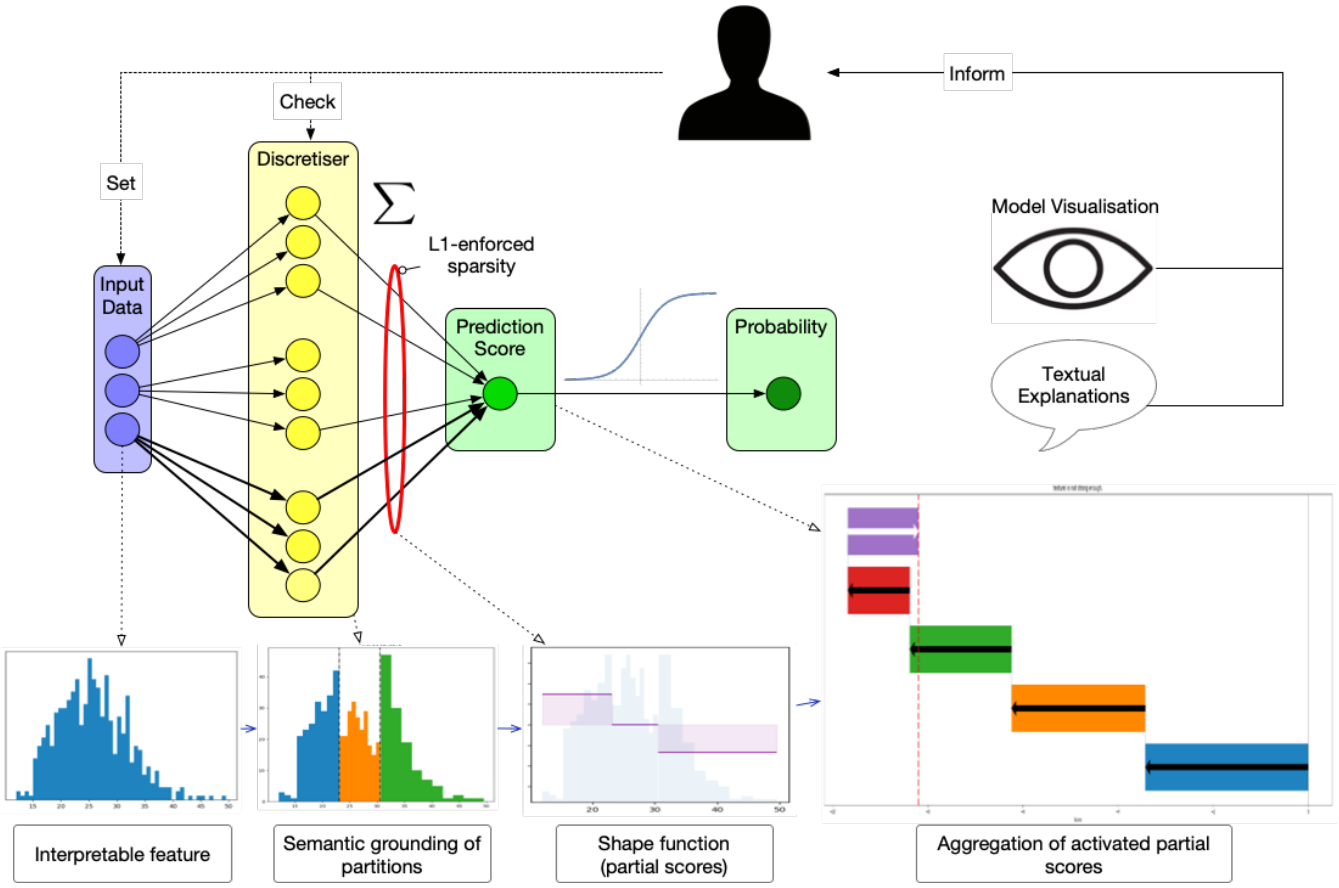


Fig. 1. A pictorial description of the proposed architecture.

methods have been developed in the field of Natural Language Generation [20]. Here, the textual generation process follows a template-based approach. Contributions are first divided into evidence and counter-evidence for the target class. They are then presented in an ordered way sorted by the absolute magnitude of their partial score so that more important features are presented first (see an illustrative example in Fig. 4, which will be described in depth in the next section). Then, the discretiser component gives us the option of using semantic labels to give a qualitative description of the magnitude of the feature. For example, if a feature had three bins (like in Fig. 1) we could establish that *for that particular feature* we assign one semantic label out of [*low*, *medium*, *high*]; with the meaning of this semantic label tied to the distribution of data. However, a given explanation can be excessively long due to many features contributing very little to the overall score. Accordingly, to keep the explanation shorter and relevant for the user, we have implemented a compression algorithm that groups small partial scores into a score called "other reasons". The values are sorted by absolute value and then are progressively accumulated, until a threshold is met (while ensuring that the absolute magnitude of the resulting aggregation is smaller than the smallest remained partial score). The rate of compression can be set by the user, allowing the user to focus first on what is most relevant and then giving him/her the option of delving into the details if needed.

It is important to note that even though the additive nature of the Prometheus model might qualitatively resemble SHAP, it is significantly different from it. This is because additive coefficients in Prometheus constitute in themselves the interpretable model. By contrast, SHAP provides these coefficients as a post-hoc explanation. Moreover Shapley values describe $f(\mathbf{x}) - \mathbb{E}_{\forall x}[f(\mathbf{x})]$ while for Prometheus they describe precisely $f(\mathbf{x})$. It is nonetheless possible to make use of the visualisation toolbox provided by the SHAP package (while keeping in mind that we are visualising something radically different).

IV. Experiments

We have empirically assessed both the accuracy and complexity (as a proxy for explainability) of the Prometheus XAI model. As humans we have a limited cognitive capacity, thus between two systems of equal performances we would tend to find more interpretable a simple one. We considered the Breast Cancer dataset (taken from the UCI machine learning data repository [21]). This is a relatively small dataset (569 instances, 30 features) where explainability, as is expected in

medical diagnosis, is an important concern. All features are numerical and there are no missing values. The task consists of predicting if a given cancer is benign (357 instances) or malign (212 instances) [6].

For comparison purposes, in addition to Prometheus, the following models are generated[4]: (i) EBMs; (ii) LM-L1 (with L1 regularisation chosen with 4-fold cross-validation); (iii) J48 decision tree; (iv) REPTree; (v) RandomTree; (vi) Fuzzy Hoeffding Decision Tree (FHDT) (vii) Fuzzy Unordered Rule Induction Algorithm (FURIA); and (viii) Random Forest (RF).

Figures 2 and 3 show the interpretability-accuracy trade-off of models which are deemed as interpretable[5]. Every measure is estimated with stratified 10-fold cross-validation. Classification performance is measured in terms of the area under the receiver operating characteristic curve (ROC AUC) in Fig. 2. Classification performance is measured in terms of F1 Score in Fig. 3. In both cases, the structural complexity of models is computed as the total rule length (TRL)[6]. In these figures, Prometheus emerges as the most accurate model, at the cost of higher structural complexity. Moreover, it performs reasonably well (ROC AUC = 0.988; F1 Score = 0.949) compared to the black-box model RF (ROC AUC = 0.991; F1 Score = 0.968). However, EBM (ROC AUC = 0.992; F1 Score = 0.971) achieves even better performance, which can be attributed to its higher flexibility due to the high number of shallow trees, which allow the modelling of a more complex shape function, and to the capability of modelling higher-order interactions. In addition, this high performance comes at the cost of increased model complexity (TRL=7680), i.e., the user has to deal with a more complex shape function representation for inspecting the model, what in practice jeopardizes intelligibility.

Regarding explainability, the prediction of a single instance consists of the summation of partial scores activated when a feature lies in a certain bin. This can be displayed in different ways (see Fig. 4). For example, the list of activated rules is given on top, a waterfall plot is given in the centre, and a linguistic explanation is given at the bottom. In addition, Fig. 5 shows the same example with a higher compression rate, i.e., with a less detailed and rougher explanation.

On the other hand, Figs. 6, 7, and 8 show the Shapely values associated with models LM-L1, RF, and EBM, respectively. These graphs provide similar information to that given by Prometheus: the final output is laid down as a sum of terms. The main difference is that this sum of terms amounts to the difference between the expected value of the function and the specific value of the instance in SHAP, while the sum directly explains the $f(x)$ in Prometheus. Moreover, it is important to notice how SHAP is a post-hoc method, thus it does not directly reflect the underlying computation. The explanation of Prometheus is instead exactly the description of its internal rationale. The explanations given by these different models show some inconsistencies, and this can be misleading and jeopardise user confidence. For example, *worst perimeter* is

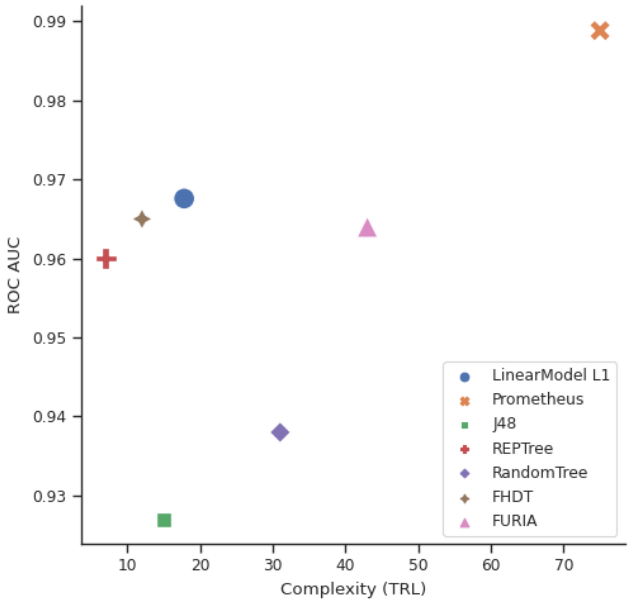


Fig. 2. The trade-off between classification performance (ROC AUC) and complexity (TRL); computed using stratified 10-fold cross-validation.

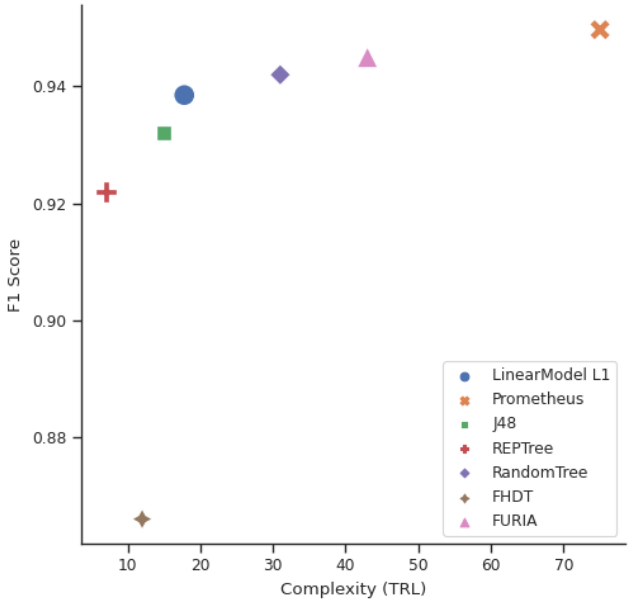


Fig. 3. The trade-off between classification performance (F1 Score) and complexity (TRL); computed using stratified 10-fold cross-validation.

[4]We use the implementation of EBMs and LM-L1 in Python InterpretML and scikit-learn packages, respectively. We use the implementation of J48, REPTree, RandomTree, FHDT, FURIA and RF in ExpliClas [8], [9].

[5]EBM is not included in the figures because it uses a discretiser with 128 bins what yields huge structural complexity (TRL=7680) and makes impossible any linguistic interpretation of the model.

[6]TRL counts the number of premises and conclusions in a rule list like the one provided by FURIA or FHDT; TRL counts the number of nodes in a tree (like J48, REPTree or RandomTree); TRL counts the number of non-zero coefficients multiplied by 2 in an LM; TRL equals 2 x number of bins x number of features in Prometheus and EBM.

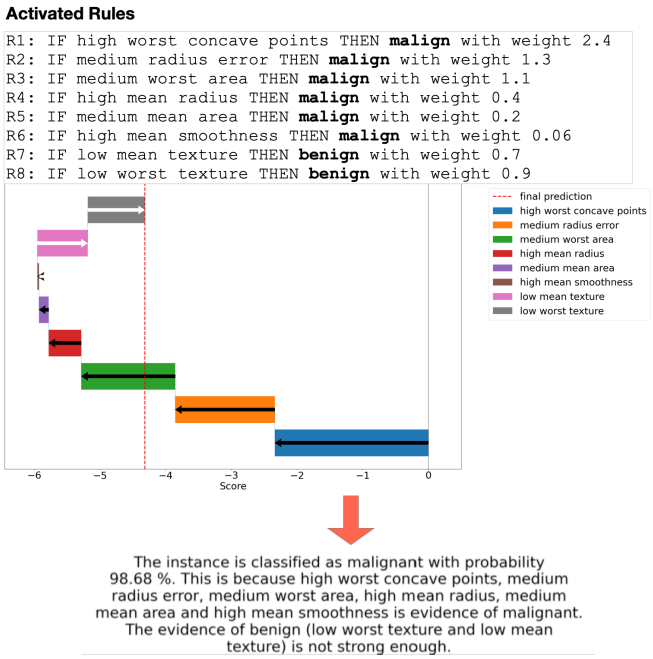


Fig. 4. A full local factual explanation provided by Prometheus.

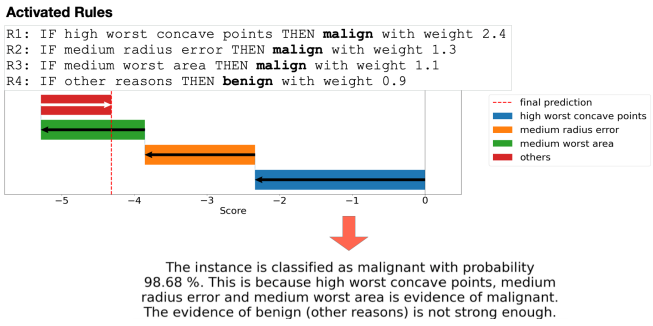


Fig. 5. Illustrative example of compressed explanation.

evidence for *benign* in case of LM-L1 but it is for *malign* in case of RF and EBM. Moreover, this feature is not highlighted among the most relevant ones by Prometheus.

In addition, the linguistic explanations provided by Prometheus can be compared to those given by ExpliClas:

- **J48**: *Diagnosis* is *malign* because *mean concavity* and *worst area* are *medium*.
- **REPTree**: *Diagnosis* is *malign* because *worst perimeter* is *medium*.
- **RandomTree**: *Diagnosis* is *malign* because *mean area*, *mean smoothness*, *area error* and *worst concavity* are *medium*.
- **FHDT**: We have a high confidence in the classification result. It is very likely that diagnosis is *malign*. There is also a medium chance that it is *benign*. On balance *malign* is more likely, because in accordance with rule 7 *concavity error* is *low* and *worst concave points* is *high*.
- **FURIA**: We have a high confidence in the classification result. It is very likely that diagnosis is *malign* because *worst concave points* is *high* and *worst radius* is *medium*.

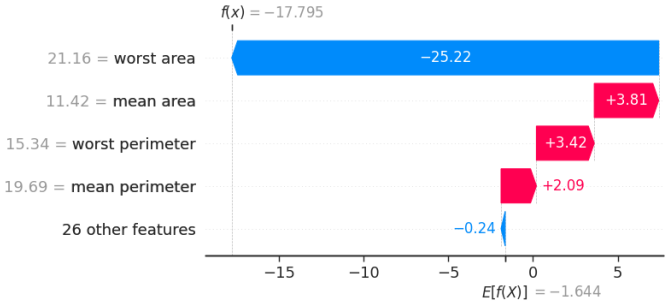


Fig. 6. Local explanation given by SHAP for LM-L1 model.

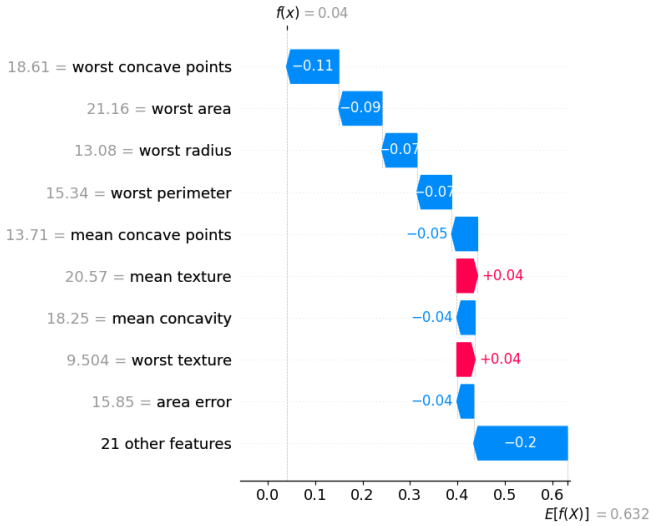


Fig. 7. Local explanation given by SHAP for RF model.

As we can see above, different models are supported by different rationale and provide consistent but different complementary explanations. In the case of fuzzy classifiers (FHDT and FURIA), ExpliClas begins explanations with a sentence related to the confidence in the classification result which emerges from the rule firing degree associated with the winner rule. This is similar to how Prometheus begins each textual explanation (see Fig. 5) with a sentence verbalising the probability associated with the given classification.

ExpliClas also provides users with factual explanations of the inferred class but nothing is said about alternative classes. This is because explanations verbalise the information contained in the activated branch of a tree or the winner rule in a fuzzy classifier. On the contrary, Prometheus provides users with contrastive explanations, which verbalise which features are evidence for and against the inferred class.

Another commonality shared by ExpliClas and Prometheus is that they provide users with a graphical output along with the list of related rules. In the case of the illustrative example under consideration, FURIA fires rules:

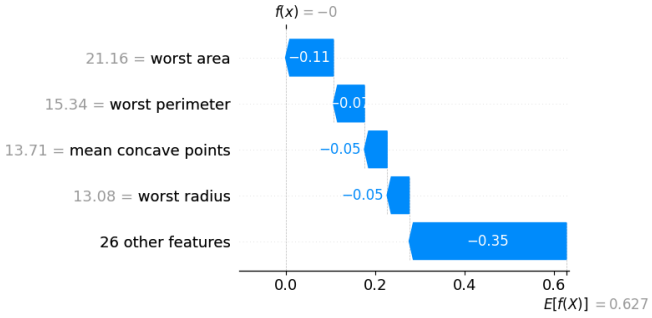


Fig. 8. Local explanation given by SHAP for EBM model.

- R1: IF *worst radius* in [16.25, 16.82, inf, inf] and *worst concave points* in [0.1452, 0.1456, inf, inf] THEN *Diagnosis* is *malign* with CF = 0.99
- R2: IF *worst area* in [947.9, 967, inf, inf] and *worst fractal dimension* in [0.06469, 0.06515, inf, inf] THEN *Diagnosis* is *malign* with CF = 0.99

However, FURIA rules are not easily verbalised, so they need to be linguistically approximated by ExpliClas as:

- R1: IF *worst radius* is *medium* and *worst concave points* is *high* THEN *Diagnosis* is *malign*
- R2: IF *worst area* is *medium* and *worst fractal dimension* is *medium* THEN *Diagnosis* is *malign*

In the case of ExpliClas, explanations only pay attention to the winner rule (R1 in this example; because the two rules are activated with the same degree and then the first rule is selected). On the contrary, all listed rules contribute to the explanation elaborated by Prometheus.

V. Conclusions

In this paper, we have introduced Prometheus, an interpretable model that can produce both textual and visual explanations related to tabular data. This model can be seen as a weighted RBS with sum aggregation at inference level. We measured reasonable performances on classification metrics, beating LMs and DTs. In addition, the model compares favourably to RFs and EBMs, though its accuracy is slightly lower. Prometheus was designed for supporting linguistic explanations with different degrees of details. As future work, we plan exhaustive experimentation with more benchmark datasets. In addition, we plan to explore more sophisticated natural language generation techniques, multiclass classification and measuring the impact of explainability with intrinsic and extrinsic human evaluation. For the sake of reproducibility, Prometheus (as well as other related XAI tools) is available at https://gitlab.citius.usc.es/jose.alonso/xai.

Acknowledgements

Jose M. Alonso-Moral is a *Ramon y Cajal* Researcher (RYC-2016-19802). This work is funded by the Spanish Ministry of Science, Innovation and Universities (grant RTI2018-099646-B-I00), the Galician Ministry of Education, University and Professional Training (grants ED431F 2018/02, ED431C 2018/29, and ED431G2019/04), and the NL4XAI project which has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 860621.

References

[1] E. Beall, "Hesiod's prometheus and development in myth," *Journal of the History of Ideas*, vol. 52, no. 3, pp. 355–371, 1991.
[2] T. Miller, "Explanation in artificial intelligence: Insights from the social sciences," *Artificial Intelligence*, vol. 267, pp. 1–38, 2019.
[3] J. M. Alonso et al., "Interactive natural language technology for explainable artificial intelligence," in *1st Workshop on Foundations of Trustworthy AI integrating Learning, Optimisation and Reasoning (TAILOR), European Conference on Artificial Intelligence (ECAI), LNAI 12641*. Springer, 2021, pp. 63–70.
[4] E. Mariotti, J. M. Alonso, and A. Gatt, "Towards Harnessing Natural Language Generation to Explain Black-box Models," in *2nd Workshop on Interactive Natural Language Technology for Explainable Artificial Intelligence*. Dublin, Ireland: ACL, 2020, pp. 22–27.
[5] J. M. Alonso, C. Castiello, L. Magdalena, and C. Mencar, *Explainable Fuzzy Systems. Paving the Way from Interpretable Fuzzy Systems to Explainable AI Systems*. Springer International Publishing, 2021.
[6] M. Patrício et al., "Using Resistin, glucose, age and BMI to predict the presence of breast cancer," *BMC Cancer*, vol. 18, pp. 1–8, 2018.
[7] A. Barredo Arrieta et al., "Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI," *Information Fusion*, vol. 58, pp. 82–115, 2020.
[8] J. M. Alonso and A. Bugarín, "Expliclas: Automatic generation of explanations in natural language for weka classifiers," in *IEEE International Conferences on Fuzzy Systems*, 2019, pp. 1–6.
[9] J. M. Alonso, P. Ducange, R. Pecori, and R. Vilas, "Building explanations for fuzzy decision trees with the ExpliClas software," in *IEEE World Congress on Computational Intelligence*, 2020, pp. 1–8.
[10] F. Hohman, A. Srinivasan, and S. M. Drucker, "TeleGam: Combining Visualization and Verbalization for Interpretable Machine Learning," in *IEEE Visualization Conference (VIS)*, 2019, pp. 151–155.
[11] Y. Lou, R. Caruana, and J. Gehrke, "Intelligible models for classification and regression," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2012, pp. 150–158.
[12] A. Goldstein et al., "Peeking Inside the Black Box: Visualizing Statistical Learning With Plots of Individual Conditional Expectation," *Journal of Computational and Graphical Statistics*, vol. 24, pp. 44–65, 2015.
[13] B. Ustun and C. Rudin, "Supersparse Linear Integer Models for Optimized Medical Scoring Systems," *Machine Learning*, vol. 102, no. 3, pp. 349–391, 2016.
[14] Y. Lou et al., "Accurate intelligible models with pairwise interactions," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2013, pp. 623–631.
[15] W. J. E. Potts, "Generalized additive neural networks," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 1999, pp. 194–200.
[16] M. T. Ribeiro, S. Singh, and C. Guestrin, "Why Should I Trust You?: Explaining the Predictions of Any Classifier," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2016, pp. 1135–1144.
[17] S. M. Lundberg and S.-I. Lee, "A Unified Approach to Interpreting Model Predictions," in *Advances in Neural Information Processing Systems (NIPS)*, 2017, pp. 1–10.
[18] S. M. Lundberg et al., "From local explanations to global understanding with explainable AI for trees," *Nature Machine Intelligence*, vol. 2, pp. 56–67, 2020.
[19] D. Slack et al., "Fooling LIME and SHAP: Adversarial Attacks on Post hoc Explanation Methods," in *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, 2020, pp. 180–186.
[20] A. Gatt and E. Krahmer, "Survey of the state of the art in natural language generation: Core tasks, applications and evaluation," *Journal of Artificial Intelligence Research*, vol. 61, pp. 65–170, 2018.
[21] D. Dua and C. Graff, "UCI Machine Learning Repository," 2017, University of California, Irvine, School of Information and Computer Sciences. [Online]. Available: http://archive.ics.uci.edu/ml

Modelando el comportamiento del consumidor con el modelo de 2-tuplas lingüísticas y una heurística basada en el conocimiento de marca

Jesús Giráldez-Cru*, Manuel Chica*†, Oscar Cordón* and Francisco Herrera*
* *Departamento de Ciencias de la Computación e Inteligencia Artificial (DECSAI)*
Instituto Andaluz Interuniversitario en Data Science and Computational Intelligence (DaSCI)
Universidad de Granada
{jgiraldez,manuelchica}@ugr.es {ocordon,herrera}@decsai.ugr.es
† *School of Electrical Engineering and Computing, The University of Newcastle (Australia)*

***Resumen*—Los modelos basados en agentes (ABM) son un paradigma de simulación para modelar sistemas complejos mediante la definición del comportamiento heterogéneo de cada uno de sus individuos en una aproximación *bottom-up*. En este trabajo, se presenta un ABM para marketing donde las percepciones de los consumidores son modeladas usando el modelo lingüístico 2-tuplas. Estas variables representan las opiniones que los consumidores tienen sobre las diferentes características de cada producto, las cuales guían sus decisiones (p. ej., precio o calidad). Al contrario que los valores exactos, las variables lingüísticas difusas son una representación realista de estos aspectos cualitativos. En nuestro ABM, los agentes usan una heurística de toma de decisiones para seleccionar un producto, la cual está basada en estas percepciones y en una regla probabilística de maximización de utilidad. Este proceso requiere una agregación difusa de las percepciones de cada producto, guiada por una serie de pesos asociados a los *drivers* del agente consumidor e implementada mediante el operador promedio ponderado de 2-tuplas. Adicionalmente, hay productos en el mercado que no son conocidos por algunos consumidores. En nuestro ABM, esta información se modela aplicando un filtro de conocimiento de marca en la heurística de toma de decisiones. De esta manera, los agentes consumidores sólo pueden elegir aquellos productos que les son conocidos. Los resultados de nuestro análisis experimental muestran que nuestra representación realista de las percepciones del consumidor es más precisa que otros métodos existentes.**

***Index Terms*—comportamiento del consumidor, marketing, heurísticas de compra, reconocimiento de marca, información y toma de decisiones lingüística difusa, 2-tuplas, modelos basados en agentes.**

I. Introducción

El principal objetivo cuando se modela un mercado es entender las reglas que lo gobiernan, para analizar posteriormente el resultado de posibles escenarios. Por tanto, es crucial entender y predecir las compras de los consumidores. En los enfoques clásicos, estas decisiones se infieren comúnmente de variables globales en un esquema *top-down*. El principal inconveniente de este paradigma es la incapacidad de representar comportamientos heterogéneos de los consumidores, y los eventos emergentes de éstos. Por esta razón, estos enfoques clásicos normalmente resultan en representaciones imprecisas de la realidad del mercado [1], [2].

Un enfoque alternativo a estos enfoques clásicos consiste en estudiar el comportamiento complejo del mercado como resultado de una agregación *bottom-up* de las decisiones de cada uno de sus consumidores [2], [3]. Para ello, los modelos basados en agentes (ABM) [4], [5] proveen una infraestructura adecuada. ABM es una técnica de modelado descriptivo (una agregación de muchas decisiones individuales de cada agente) que ayuda al diseñador y al profesional de marketing a entender mejor el mercado y su comportamiento. En la mayoría de los casos, modelar comportamientos individuales es más simple y más preciso que modelar el comportamiento del sistema completo por reglas globales. Los ABM se han aplicado con éxito en otras áreas tan diversas como economía [6], política [7] y sistemas de confianza social [8].

La mayoría de los ABM para marketing existentes representan las opiniones del consumidor usando valores numéricos [9]. Ésta es una representación poco realista de este tipo de información cualitativa. Además, las opiniones del consumidor se definen normalmente usando datos de cuestionarios disponibles para la compañía, los cuales son comúnmente respondidos en términos lingüísticos. Por tanto, manejar valores numéricos requiere el preproceso adicional para transformar las respuestas lingüísticas de los cuestionarios en datos exactos, con lo que el diseño del modelo es más complejo y suele provocar una pérdida de información en la representación.

En este trabajo, se desarrolla un ABM para modelar mercados virtuales con una representación realista de las percepciones del consumidor, la cual está basada en variables lingüísticas difusas [10]–[12]. Esta información permite dar valores difusos (p.ej., a cada una de las características de los productos (p.ej., precio, calidad o gusto) [13]. En concreto, representamos estas percepciones usando el modelo lingüístico 2-tuplas [14], que consisten en un par formado por una etiqueta lingüística y un desplazamiento simbólico. Como se explica en [13], esta representación solventa los inconvenientes de otros enfoques lingüísticos difusos ordinales [15] ya que permite asignar valores diferentes a dos variables que tengan la misma etiqueta lingüística (teniendo dos valores diferentes en el desplazamiento simbólico). Además, esta representación

es sustancialmente más realista que otras representaciones existentes y estándares como los valores numéricos u otro tipo de valores exactos.

Dado que cada agente tiene una percepción sobre cada característica de los productos del mercado, se propone una heurística de toma de decisiones difusa que agrega dichas percepciones lingüísticas difusas para calcular una valoración global de cada producto para el consumidor y elegir uno de ellos, lo cual simula el proceso de compra de los consumidores. Dado que las percepciones del consumidor se representan con variables lingüísticas difusas y tienen un vector de pesos asociado a cada variable (*driver*), su agregación se puede realizar con el operador promedio ponderado para 2-tuplas [14].

De este modo, el ABM para el análisis de comportamiento de consumidor con representación lingüística difusa presentado en [13] representa adecuadamente varias características del comportamiento real del consumidor, pero no considera mecanismos más complejos como procesos de boca a boca en redes sociales u otro tipo de interacciones complejas entre agentes. Para avanzar en el diseño de un ABM realista para marketing, en este trabajo se introducen dos importantes modificaciones con respecto a dicho modelo. Primero, las decisiones del consumidor raramente son completamente deterministas. En nuestro sistema, este comportamiento se modela por medio de una función probabilística de maximización de la utilidad proporcional a las valoraciones agregadas de los productos, a diferencia de la estrategia plenamente determinista usada en [13]. Segundo, los consumidores no siempre conocen todas las marcas en el mercado. En el ABM propuesto en este trabajo, se incluye esta información definiendo las marcas de las que cada agente tiene conocimiento. Este tipo de información está disponible normalmente en datos de encuestas sobre el consumidor facilitado por agencias de marketing. Nuestra heurística de toma de decisiones implementa un filtro de conocimiento de marca, de forma que los agentes consumidores sólo pueden comprar productos que conocen.

El modelo propuesto se evalúa nuestro sistema en un caso de estudio de marketing real, comparando su rendimiento con el de otras representaciones tradicionales de las percepciones del consumidor, analizando su precisión con respecto a datos reales que representan las ventas en este mercado. El análisis experimental muestra que nuestro modelo es más preciso.

El resto de este trabajo se organiza así: en la Sec. II se describen conceptos preliminares; el ABM propuesto se presenta en Sec. III; en la Sec. IV se presenta una evaluación experimental del modelo; y finalmente se concluye en la Sec. V.

II. Contexto

II-A. Representación numérica de las preferencias del consumidor usando datos de encuestas

Los sistemas ABM para marketing normalmente requieren de una definición de las percepciones del consumidor para cada marca del mercado, con el fin de simular un mercado virtual realista. Estas percepciones se obtienen de datos de encuestas y estudios de salud de marca de consumidores reales, facilitados por renombradas consultoras de marketing como Kantar Millward-Brown [16]. Estos datos se estructuran en forma de encuestas incluyendo conjuntos de respuestas a una seria de preguntas sobre las marcas [17].

En algunos casos, las respuestas de las encuestas se recogen directamente en la misma escala requerida por el ABM, un valor real en el intervalo $[0, 10]$, representando 0 la percepción más negativa, 5 una percepción neutral, y 10 la percepción más positiva. Sin embargo, la situación más habitual es que las respuestas de las encuestas muestren una naturaleza lingüística. En estos casos, se requiere un preprocesado manual de las respuestas para traducirlas a la escala [0,10], con la consecuente *pérdida de información*. Por tanto, este problema se aborda mejor trabajando directamente con las valoraciones lingüísticas siguiendo un enfoque lingüístico difuso en lugar de transformarlas a valores numéricos.

II-B. Variables lingüísticas

Las variables lingüísticas [10]–[12] toman como valores palabras o frases en el lenguaje natural. Se usan en *enfoques lingüísticos difusos*, donde el problema requiere manejar aspectos cualitativos [15]. Éste es un requerimiento típico en muchos contextos, donde la representación de la información más directa y realista es claramente el lenguaje natural.

En el *enfoque lingüístico difuso ordinal*, un caso especial de enfoques lingüísticos difusos, las variables lingüísticas toman valores de un conjunto predefinido y totalmente ordenado de *etiquetas lingüísticas* $S = \{s_0, \ldots, s_g\}$ de tamaño finito $|S| = g + 1$. Se considera la definición convencional de conjunto ordenado donde $\forall s_i, s_j \in S \,.\, s_i \leq s_j \Leftrightarrow i \leq j$.

En este trabajo, consideramos funciones de pertenencia triangulares para las variables lingüísticas. En la Figura 1 se representa un ejemplo de esta pertenencia difusa. En concreto, se usa una función de pertenencia triangular definida en el intervalo $[0, 1]$ para un conjunto de 5 etiquetas lingüísticas $\{muyMala, mala, neutra, buena, muyBuena\}$ de la variable lingüística.

El manejo de variables lingüísticas difusas normalmente requiere agregar su información, es decir, los valores de estas variables. Un enfoque común es aproximar estos valores lingüísticos con valores numéricos correspondientes a su índice en el conjunto ordenado de etiquetas $\{0, 1, \ldots, g\}$, agregarlos con métodos comunes, obteniendo un valor real intermedio $\beta \in [0, g]$, y finalmente aproximar este valor intermedio a una etiqueta lingüística del conjunto original [18]. Para ello, se definen dos operadores para aproximar variables lingüísticas a números y *vice versa*:

Definición 1 (Aproximaciones Numérico-Lingüísticas del modelo simbólico lingüístico computacional ordinal)**.** *Dado un número real $\beta \in [0, g]$ y una etiqueta lingüística s_k, siendo k la posición de la etiqueta s_k en el conjunto ordenado de etiquetas lingüísticas $S = \{s_0, \ldots, s_g\}$ del que la variable lingüística v toma valores, se definen las siguientes*

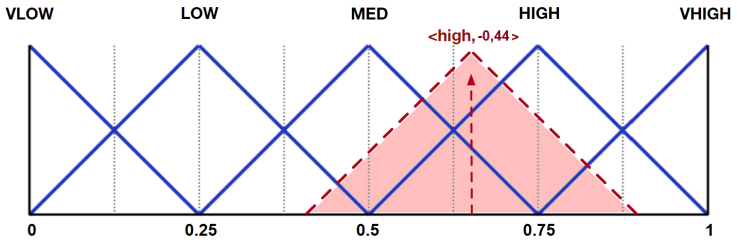


Figura 1. Función de pertenencia triangular para variables lingüísticas en el intervalo $[0,1]$ para el conjunto de etiquetas lingüísticas $\{muyMala, mala, neutra, buena, muyBuena\}$. Se incluye también la 2-tupla correspondiente a la aproximación $\Delta(2{,}56) = \langle buena, -0{,}44\rangle$.

aproximaciones numérico-lingüísticas $\Delta' : S \rightarrow \{0, ..., g\}$ *y* $\Delta : [0, g] \rightarrow S$ *como:*

$$\Delta'(s_k) = k$$
$$\Delta(\beta) = s_k \text{ s.t. } \forall_{s_i \in S \wedge i \neq k} \; |\Delta'(s_k) - \beta| < |\Delta'(s_i) - \beta| \vee \exists_{s_i \in S \wedge i = k+1} \; |\Delta'(s_k) - \beta| = |\Delta'(s_i) - \beta|$$

En resumen, el valor numérico asociado a una etiqueta lingüística s_k corresponde con su posición en el orden del conjunto y la etiqueta asociada al valor numérico β es la más cercana de acuerdo a los valores numéricos de las etiquetas en la escala $[0, g]$. En el caso de que β quede justo en el punto medio entre dos etiquetas, por convenio se asigna a la menor en el orden.

II-C. El operador de agregación promedio ponderado para variables lingüísticas

Existe una amplia variedad de operadores de agregación en la literatura que han sido adaptados para trabajar con variables lingüísticas [18], destacando el operador OWA [19] debido a su gran versatilidad. En nuestro caso, cada agente consumidor tiene un peso asociado a cada *driver*, previamente especificado, que representa sus preferencias de compra, por lo que emplearemos el operador promedio ponderado.

Definición 2 (Promedio ponderado para variables lingüísticas [18])**.** *Sean* $(a_1, \ldots, a_m)$ *una serie de etiquetas lingüísticas a agregar, con* $a_i \in S$*, y sea* $W = \{w_1, \ldots, w_m\}$ *el conjunto de pesos asociado. El operador promedio ponderado de variables lingüísticas se define como:*

$$\phi(A, W) = \Delta\left(\frac{\sum_{i=1}^{m} \Delta'(a_i) \cdot w_i}{\sum_{i=1}^{m} w_i}\right)$$

donde las funciones Δ *y* Δ' *son las aproximaciones numérico-lingüísticas definidas anteriormente.*

En nuestro caso el vector de pesos W cumple que $w_i \in [0, 1]$ y $\sum_i w_i = 1$, por lo que tenemos una combinación lineal y el denominador vale 1.

II-D. El modelo de representación de 2-tuplas lingüísticas

Un inconveniente del enfoque anterior es la pérdida de información causada por la agregación de etiquetas lingüísticas. Para resolver este problema, en [14] se propuso la *representación de 2-tuplas lingüísticas*.

Definición 3 (2-tupla lingüística [14])**.** *Una 2-tupla lingüística es un par* $\langle s_k, \alpha\rangle$*, en el que* $s_k \in S$ *es una etiqueta lingüística y* $\alpha \in [-0{,}5, 0{,}5)$ *es un desplazamiento simbólico que especifica la traslación de la función de pertenencia que representa la etiqueta lingüística* s_k *más cercana si la información lingüística resultante de un cálculo simbólico no corresponde exactamente con una etiqueta del conjunto. El conjunto de 2-tuplas asociado con* S *se define como* $\bar{S} = S \times [-0{,}5, 0{,}5)$.

Definición 4 (Aproximaciones Numérico-Lingüísticas de 2-tuplas lingüísticas [14])**.** *Dado un número real* $\beta \in [0, g]$ *y una etiqueta lingüística* $s_k \in S$*, se definen las siguientes aproximaciones numérico-lingüísticas* $\Delta' : S \rightarrow \{0, ..., g\}$ *y* $\Delta : [0, g] \rightarrow \bar{S}$ *como:*

$$\Delta'(\langle s_k, \alpha\rangle) = k + \alpha$$
$$\Delta(\beta) = \langle s_k, \alpha\rangle, \text{donde } k = round(\beta) \text{ y } \alpha = \beta - k$$

La función $round$ *devuelve el entero* $k \in \{0, ..., g\}$ *más cercano a* β.

De este modo, una 2-tupla lingüística se vincula a un valor numérico equivalente β en el intervalo de granularidad de S, $[0, g]$, y ese valor se obtiene a partir del valor numérico de la etiqueta y el del desplazamiento. Por otro lado, un valor numérico β se asocia a una 2-tupla compuesta por la etiqueta lingüística más cercana según los valores numéricos de las etiquetas en la escala $[0, g]$ y al desplazamiento necesario para hacer coincidir el valor de dicha etiqueta con el de β.

Por tanto, el operador de agregación promedio ponderado de la Definición 2 se puede aplicar directamente a 2-tuplas lingüísticas usando las aproximaciones numérico-lingüísticas de la Definición 4 [14]. Un ejemplo de estas aproximaciones se muestra en la Figura 1, donde se ve como $\Delta(2{,}56) = \langle buena, -0{,}44\rangle$.

III. Modelo basado en agentes con información lingüística difusa y conocimiento de marca

En esta sección se define el ABM que simula el comportamiento de un mercado con percepciones del consumidor lingüísticas y una heurísticas de toma de decisión lingüística. En nuestro modelo, usamos variables lingüísticas difusas para representar los diferentes aspectos de cada marca o producto (p.ej., *precio*, *calidad*, *confort*, ...). A estos aspectos se les llama *drivers* dado que guían las elecciones de los

consumidores. En concreto, usamos 2-tuplas lingüísticas para almacenar las percepciones sobre los *drivers* que gobiernan el comportamiento del mercado.

En nuestro modelo, los agentes representan consumidores que llevan a cabo un proceso de toma de decisiones para seleccionar un producto entre las marcas disponibles. Este proceso se lleva a cabo según sus propias percepciones y la valoración que le dan a cada marca. La población de agentes se organiza en segmentos, grupos de agentes muy similares en términos de comportamiento. Todo ello nos permite simular el comportamiento del mercado y realizar predicciones en él.

III-A. Marcas

En nuestro ABM, las estrategias de toma de decisión de los consumidores únicamente se llevan a cabo entre un conjunto finito $B = \{b_1, \ldots, b_n\}$ de n marcas disponibles en el mercado. Para modelar los atributos de cada marca, se considera también un conjunto $D = \{d_1, \ldots, d_m\}$ de m *drivers*. Estos *drivers* son fijos para todas las marcas en el mercados.

III-B. Percepciones del consumidor

Todo consumidor está representado por un agente en el sistema. Cada agente tiene sus propias percepciones (positiva, neutra o negativa) sobre cada *driver* de cada marca. Para representar las preferencias de los *drivers*, se define para cada agente x un vector de pesos $W^x = [w_1^x, \ldots, w_m^x]$, tal que todos los pesos deben estar en el intervalo $[0, 1]$ y su suma debe ser igual a 1. Estos pesos representan la importancia de cada *driver* cuando un agente consumidor x realiza una decisión de compra.

Las percepciones del consumidor se modelan definiendo, para cada agente x del ABM, una matriz de percepciones P^x de dimensiones $n \times m$, donde cada elemento $p_{i,j}^x \in P^x$ representa la percepción del agente x sobre el *driver* $d_j \in D$ de la marca $b_i \in B$. En nuestro modelo, estas percepciones se representan usando 2-tuplas lingüísticas, todas ellas tomando valores de un conjunto ordenado de etiquetas lingüísticas común (ver la Definición 3). Esto nos permite representa la visión cualitativa del consumidor sobre cada marca.

III-C. El conocimiento de marca del consumidor

En un mercado real, cada consumidor puede no tener conocimiento de ciertas marcas. Para modelar esta información, se define en nuestro modelo el conocimiento de marca de cada agente. En concreto, para cada agente x, se define un vector A^x de n variables Booleanas, donde $a_i^x \in A^x$ representa si el agente x tiene conocimiento de la marca $b_i \in B$.

III-D. Agentes consumidores

Basados en la caracterización de las marcas y de los consumidores presentada anteriormente, ahora pasamos a definir los agentes consumidores. Nótese que esta definición representa el estado mental de los consumidores, es decir, su conocimiento sobre el mercado y sus percepciones sobre los productos disponibles.

Definición 5 (Agente consumidor)**.** *Un agente consumidor x se define como la tupla $\langle A^x, W^x, P^x \rangle$, donde A^x, W^x y P^x son respectivamente el vector de conocimiento de marca, el vector de pesos de drivers y la matriz de percepciones en 2-tuplas lingüísticas descritos anteriormente.*

III-E. Heurística de toma de decisión

El proceso de toma de decisión difusa de cada agente consiste en seleccionar una de las marcas disponibles en el ABM en función de sus percepciones sobre los *drivers*. Estas decisiones simulan decisiones de compras de los consumidores en el mercado. El proceso se divide en dos pasos: (i) la agregación de las valoraciones de cada marca, y (ii) la selección de una marca. En el primer paso, el agente necesita agregar las percepciones de todos los *drivers* de cada marca. Esta agregación se calcula usando el operador de agregación promedio para 2-tuplas de la forma:

Definición 6 (Valoración de marca)**.** *Dados un agente consumidor x y una marca b_i, se define la valoración de marca $as(x, b_i)$ como la agregación de sus percepciones para este marca calculadas con el operador de agregación promedio ϕ:*

$$as(x, b_i) = \phi(P_i^x, W^x) = \Delta(W^x \cdot \Delta'((P_i^x)^T))$$

donde $P_i^x = [p_{i,1}^x, \ldots, p_{i,m}^x]$ es la fila i-ésima de la matriz P^x, W^x es el vector de pesos de drivers del agente x, y Δ y Δ' son las funciones de aproximación numérico-lingüísticas para 2-tuplas de la Definición 4 [1].

El segundo es la selección de una marca en función de la valoración que el agente hace sobre cada marca. En este trabajo se usa una función de maximización de la utilidad probabilística $maxUtil^P$, que asigna a cada marca una probabilidad proporcional a su valoración y elige aleatoriamente una marca utilizando una ruleta probabilística. Se usa esta función no determinista inspirada por trabajos previos en análisis de marketing y comportamiento del consumidor [20]–[22].

Definición 7 (Maximización de la utilidad probabilística)**.** *Para un agente consumidor x, la función de maximización de utilidad probabilística $maxUtil^P$ es la función probabilística de selección de marca que elige aleatoriamente una marca usando una ruleta probabilística con las siguiente probabilidades p_x para cada marca $b_i \in B$:*

$$p_x(b_i) = e^{\Delta'(as(x,b_i))} = e^{W^x \cdot \Delta'((P_i^x)^T)}$$

Sin pérdida de generalidad, estas probabilidades no normalizadas se pueden normalizar simplemente como $p_x(b_i) = p_x(b_i) / \sum_{b_j \in B} p_x(b_j)$.

IV. Evaluación experimental del modelo

En esta sección se presenta una evaluación experimental del ABM para marketing con representación lingüística difusa en un caso de estudio real. En concreto, se presenta una comparativa entre nuestro ABM y otros dos modelos que

[1] Por simplicidad, se han sobrecargado las funciones Δ y Δ' para vectores de la forma: $\Delta(B) = [\Delta(b_i)]_{1 \leq i \leq n}$ y $\Delta'(A) = [\Delta'(a_i)]_{1 \leq i \leq n}$.

únicamente difieren en la representación de las percepciones (el resto del modelo se mantiene sin ningún otro cambio).

Por una parte, se compara nuestro ABM frente a un modelo con representación numérica de las percepciones, en el intervalo $[0, 10]$. Por otra, se compara nuestro ABM frente a un modelo simbólico lingüístico ordinal donde las percepciones se representan usando etiquetas lingüísticas (usando el mismo conjunto de etiquetas que en las 2-tuplas).

Los resultados de estos tres modelos se validan usando datos reales que representan las ventas en un mercado concreto y se analiza el rendimiento de los tres modelos midiendo su precisión en términos de predicción en ventas (agregadas para todos los agentes consumidores en el ABM).

IV-A. Descripción de las condiciones de simulación del ABM

Para la inicialización de los agentes se usan segmentos de consumidores, es decir, grupos de consumidores con un comportamiento *similar*. En nuestro ABM, todos los agentes del mismo segmento se caracterizan por los mismos pesos W^x y similares percepciones P^x, generadas aleatoriamente siguiendo una distribución normal con media igual al promedio del segmento y pequeña desviación estándar.[2] Los segmentos no introducen ningún cambio en el ABM, únicamente afectan a las percepciones de los agentes.

Para reducir la influencia de valores atípicos producidos por la inicialización aleatoria de las percepciones, cada ejecución del ABM está compuesta por un número de simulaciones Monte Carlo (MC). [3]

IV-B. Validación en un caso de estudio real para marketing

En esta subsección se presentan los resultados de la ejecución del ABM. Los resultados representan el número de elecciones en los que cada marca es elegida, considerando que cada agente únicamente lleva a cabo un único proceso de toma de decisión.

En este caso de estudio, se ejecuta nuestro ABM con 1000 agentes, inicializando sus pesos de *drivers* y sus percepciones usando estudios de marketing existentes. En los dos enfoques lingüísticos difusos, se usan como etiquetas lingüísticas el conjunto $S = \{mala, neutra, buena\}$. Este caso de estudio contiene 5 marcas y 6 *drivers*.[4]

En nuestro estudio, se realizan dos experimentos diferentes, uno usando el filtro de conocimiento de marca en la heurística de toma de decisiones y el otro sin usarlo. Cuando este filtro está activo, un agente sólo es capaz de elegir aquellas marcas sobre las que tiene conocimiento. Para desactivarlo, sencillamente se establece al 100 % el conocimiento de marca para todos los agentes y todas las marcas.

La Figura 2 muestra la comparativa de los tres modelos sin usar el filtro de conocimiento de marca (izquierda) y usándolo (derecha). Para ello, se presentan los diagramas de caja donde se representan los valores máximo, mínimo, mediana, y cuartiles primero y tercero del número de elecciones de cada marca.

Se observa que los resultados en ambos experimentos son diferentes, para cualquier representación de las percepciones del consumidor. Se puede ver, por ejemplo, el número de ventas de la *marca2* usando la representación lingüística 2-tuplas. Cuando no se considera el conocimiento de marca, este número es mucho mayor que el número de ventas cuando este filtro de conocimiento de marca está activo. Además, se pueden observar diferencias en los resultados de los tres modelos. Son especialmente significativas las diferencias de nuestro ABM con respecto al modelo con etiquetas lingüísticas. Por ejemplo, el número de consumidores que eligen la *marca5* es mucho menor en el ABM que modela las percepciones con etiquetas lingüísticas (con y sin conocimiento de marca).

Para medir la precisión de cada modelo, se calcula como estimadores del error el Error Absoluto Medio (MAE, *Mean Absolute Error*) y la Raíz del Error Cuadrático Medio (RMSE, *Root Mean Squared Error*) con respecto a datos reales (es decir, ventas reales).

Cuadro I
MAE Y RMSE DE LOS TRES MODELOS CON RESPECTO A DATOS REALES, CON Y SIN CONOCIMIENTO DE MARCA. MEJOR RESULTADO EN NEGRITA.

Estimador del error	Conocimiento de marca	Representación de las percepciones		
		Numérica	2-tuplas	Et. ling.
MAE	No	2.83	2.95	5.25
	Sí	1.78	**1.60**	2.46
RMSE	No	3.08	3.20	6.23
	Sí	2.14	**1.95**	2.98

Los valores de los estimadores MAE y RMSE para los tres modelos se reportan en el Cuadro I. Se puede observar que los modelos con representaciones de las percepciones del consumidor en formatos numéricos y 2-tuplas son mucho más precisos que el modelo con etiquetas lingüísticas. Esto es consecuencia de la representación menos expresiva usada en el modelo lingüístico difuso ordinal. Se hace énfasis en que este problema no aparece en nuestro ABM basado en la representación lingüística 2-tuplas. Además, se puede observar que el rendimiento de cada modelo es mucho peor cuando el filtro de conocimiento de marca no está activo. Esto sugiere que el concomiendo de marca es otra componente fundamental para simular un mercado con precisión. De hecho, el modelo más preciso para ambos estimadores del error (MAE y RMSE) es aquel que representa las percepciones del consumidor usando 2-tuplas lingüísticas y usando el filtro de conocimiento de marca, es decir, el modelo presentado en este trabajo.

V. CONCLUSIONES Y TRABAJO FUTURO

En este trabajo se ha presentado un modelo lingüístico difuso para representar las percepciones de los consumidores, así como una heurística de toma de decisiones que las maneja y que se usa para simular las compras de los consumidores. Todo ello se integra en un ABM para el análisis en marketing, donde los agentes representan a los consumidores del mercado.

[2]En nuestros experimentos, se usa una desviación estándar de 1,5.
[3]Cada ejecución del ABM está compuesta de 100 simulaciones MC.
[4]Los nombres de las marcas se han omitido por razones de anonimato.

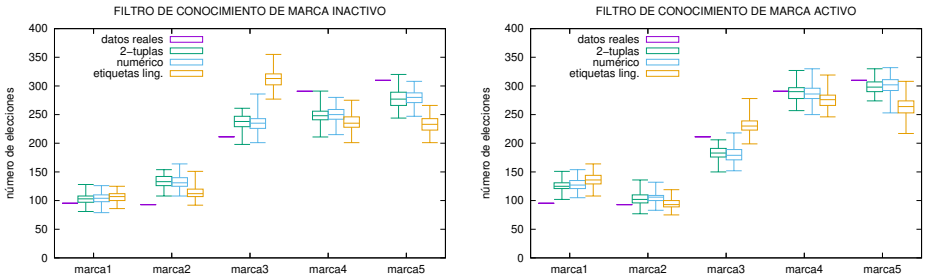


Figura 2. Comparación de las representaciones de percepciones sin usar el filtro de conocimiento de marca (izquierda) y usándolo (derecha).

Las percepciones de los consumidores son normalmente aspectos cualitativos. En nuestro modelo, se usan 2-tuplas lingüísticas, las cuales no sufren ninguna pérdida de información, incluso en el proceso de agregación de la información. En la vida real, los consumidores tienen un conocimiento de marca limitado, por lo que hemos incorporado también esta información en nuestro sistema para hacerlo más realista. Hemos definido una heurística no determinista basada en las valoraciones que cada consumidor realiza sobre cada producto para simular el proceso de toma de decisiones de compra. Se han analizado experimentalmente los resultados de nuestro ABM en un caso de estudio de marketing real a gran escala, mostrando diferencias notables en los diferentes escenarios donde únicamente se modifica la representación de las percepciones del consumidor (sin alterar sus valores), y además se ha estudiado la influencia de incluir información sobre el conocimiento de marca del consumidor en el modelo.

Como trabajo futuro, se planea extender este ABM para marketing basado en 2-tuplas lingüísticas en dos direcciones. Por una parte, se planea investigar otras heurísticas de tomas de decisión en nuestro sistema [20]–[22], adaptándolas para manejar información lingüística difusa. Por otra, se planea extender nuestro ABM para marketing incorporando comportamiento temporal para construir simulaciones con eventos discretos [3]–[5]. De esta forma, las percepciones de cada consumidor pueden cambiar a lo largo del tiempo, y por tanto, se puede analizar como esto afecta a los resultados de la toma de decisiones.

Agradecimientos

Trabajo financiado por MICIU, JdA, UGR, AEI, y ERDF (PGC2018-101216-B-I00, A-TIC-284-UGR18, PY18-4475, FJCI-2017-32420). Agradecemos a la compañía ZIO Analytics por los datos aportados, usados en este trabajo.

Referencias

[1] G. Overgoor, M. Chica, W. Rand, and A. Weishampel, "Letting the computers take over: Using AI to solve marketing problems," *California Management Review*, p. 0008125619859318, 2019.

[2] W. Rand and R. T. Rust, "Agent-based modeling in marketing: Guidelines for rigor," *International Journal of Research in Marketing*, vol. 28, no. 3, pp. 181–193, 2011.

[3] M. Chica and W. Rand, "Building agent-based decision support systems for word-of-mouth programs. a freemium application," *Journal of Marketing Research*, vol. 54, pp. 752–767, 2017.

[4] C. M. Macal and M. J. North, "Tutorial on agent-based modeling and simulation," in *Proceedings of the 37th Conference on Winter Simulation*. ACM, 2005, pp. 2–15.

[5] J. M. Epstein, *Generative social science: Studies in agent-based computational modeling*. Princeton University Press, 2006.

[6] L. Hamill and N. Gilbert, *Agent-Based Modelling in Economics*, 1st ed. Wiley Publishing, 2016.

[7] I. Moya, M. Chica, J. L. Sáez-Lozano, and O. Cordón, "An agent-based model for understanding the influence of the 11-M terrorist attacks on the 2004 spanish elections," *Knowledge-Based Systems*, vol. 123, pp. 200–216, 2017.

[8] M. Chica, R. Chiong, M. Adam, and T. Teubner, "An evolutionary game model with punishment and protection to promote trust in the sharing economy," *Scientific Reports*, vol. 9, p. 19789, 2019.

[9] M. Janssen and W. Jager, "An integrated approach to simulating behavioural processes: A case study of the lock-in of consumption patterns," *Journal of Artificial Societies and Social Simulation*, vol. 2, no. 2, 1999.

[10] L. A. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning - I," *Information Sciences*, vol. 8, no. 3, pp. 199–249, 1975.

[11] ——, "The concept of a linguistic variable and its application to approximate reasoning - II," *Information Sciences*, vol. 8, no. 4, pp. 301–357, 1975.

[12] ——, "The concept of a linguistic variable and its application to approximate reasoning - III," *Information Sciences*, vol. 9, no. 1, pp. 43–80, 1975.

[13] J. Giráldez-Cru, M. Chica, O. Cordón, and F. Herrera, "Modeling agent-based consumers decision-making with 2-tuple fuzzy linguistic perceptions," *International Journal of Intelligent Systems*, vol. 35, pp. 283–299, 2020.

[14] F. Herrera and L. Martínez-López, "A 2-tuple fuzzy linguistic representation model for computing with words," *IEEE Transactions on Fuzzy Systems*, vol. 8, no. 6, pp. 746–752, 2000.

[15] E. Herrera-Viedma, "Modeling the retrieval process for an information retrieval system using an ordinal fuzzy linguistic approach," *Journal of the American Society for Information Science and Technology*, vol. 52, no. 6, pp. 460–475, 2001.

[16] M. Brown. (2019) Brand tracking studies. [Online]. Available: http://www.millwardbrown.com/mb-global/what-we-do/brand/brand-tracking

[17] A. Charan, *Marketing analytics: a practitioner's guide to marketing analytics and research methods*. World Scientific Publishing Company, 2015.

[18] M. Delgado, J. L. Verdegay, and M. A. Vila, "On aggregation operations of linguistic labels," *International Journal of Intelligent Systems*, vol. 8, no. 3, pp. 351–370, 1993.

[19] R. R. Yager, "On ordered weighted averaging aggregation operators in multicriteria decisionmaking," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 18, no. 1, pp. 183–190, 1988.

[20] J. R. Bettman, M. F. Luce, and J. W. Payne, "Constructive Consumer Choice Processes," *Journal of Consumer Research*, vol. 25, no. 3, pp. 187–217, 1998.

[21] G. Gigerenzer and W. Gaissmaier, "Heuristic decision making," *Annual Review of Psychology*, vol. 62, no. 1, pp. 451–482, 2011.

[22] C. del Campo, S. Pauser, E. Steiner, and R. Vetschera, "Decision making styles and the use of heuristics in decision making," *Journal of Business Economics*, vol. 86, no. 4, pp. 389–412, 2016.

Generación de descripciones lingüísticas de la contaminación acústica diaria en zonas urbanas

Juan Moreno-Garcia
Escuela de Ingeniería Industrial y Aeroespacial
Universidad de Castilla-La Mancha
Toledo, Spain
juan.moreno@uclm.es

Luis Jimenez-Linares
Escuela Superior de Informática
Universidad de Castilla-La Mancha
Ciudad Real, Spain
luis.jimenez@uclm.es

Luis Rodriguez-Benitez
Escuela Superior de Informática
Universidad de Castilla-La Mancha
Ciudad Real, Spain
luis.rodriguez@uclm.es

***Abstract*—Uno de los mayores problemas que preocupan a la sociedad actual es la contaminación, distinguiéndose diferentes tipos, por ejemplo, acústica, ambiental, térmica, etc. Entre ellas, la contaminación acústica causa serios problemas a los ciudadanos porque es continua durante gran parte del día, debido a que en su mayoría es causada por el tráfico. Por otro lado, las grandes ciudades aportan una gran cantidad de datos obtenidos diariamente gracias a la sensorización derivada del concepto de "ciudades inteligentes", lo que permite visualizar la información de las zonas sensorizadas y alertar a las instituciones de los problemas y, a los ciudadanos, conocer la situación de la contaminación acústica en base a datos para poder realizar las quejas y denuncias pertinentes a las instituciones. Una forma universalmente comprensible de mostrar la información contenida en los datos capturados es la generación de descripciones lingüísticas que sinteticen la información que reside en los datos. Este trabajo presenta un método para generar descripciones lingüísticas a partir de los datos de contaminación acústica capturados por las estaciones de medición de ruido. Se presentará un método de generación de descripciones de un día que considera los periodos diarios en los que se estructuran los datos tomados de las estaciones (diurno, vespertino, nocturno y diurno completo). Para probar el método propuesto, se han utilizado los datos disponibles de la ciudad de Madrid para generar descripciones que permitan analizar la influencia de Covid-19 en la contaminación acústica.**

***Index Terms*—descripción lingüística, lógica difusa, contaminación acústica, Covid-19**

I. Introducción

Hoy en día existe una gran concienciación sobre los problemas medioambientales y esto hace que las instituciones lleven a cabo iniciativas para mejorar la situación. La contaminación es uno de los problemas ambientales que más atención recibe debido a su impacto en la salud de las personas y otros seres vivos. Entre ellos se encuentra la contaminación acústica, que llamaremos ruido, y que puede definirse como el exceso de sonido que altera las condiciones ambientales de una zona determinada. El ruido tiene graves consecuencias de todo tipo para quienes lo padecen, siendo los problemas más destacados: los problemas auditivos por estar sometidos de forma habitual a un exceso de ruido en el ambiente, los problemas de sueño (alteración del ciclo del sueño, insomnio, somnolencia durante el día, cansancio, etc.), los problemas psicológicos (irritabilidad, estrés, ansiedad, etc.) y los fisiológicos (aumento del ritmo cardíaco y respiratorio o de la presión arterial). En las grandes ciudades, las fuentes de ruido son muy diversas, siendo las más importantes el tráfico y la construcción, aunque también se debe a la actividad de las personas en la ciudad. La Organización Mundial de la Salud (OMS) recomienda no superar los 65 decibelios durante el día y los 55 durante la noche.

Además, las ciudades están aplicando el concepto de "ciudades inteligentes" para lograr una mayor sostenibilidad económica, social y medioambiental mediante el uso de las tecnologías de la información y la comunicación (TIC). La aplicación de las TIC crea infraestructuras que pretenden garantizar un desarrollo sostenible, un aumento de la calidad de vida de los ciudadanos, una mayor eficiencia de los recursos disponibles y una participación ciudadana activa. Por ello, las ciudades disponen de datos diarios procedentes de muy diversas fuentes, entre las que se encuentran las estaciones de medición de ruido, que pueden utilizarse para lograr la sostenibilidad deseada. A partir de ellos, se puede obtener información sobre el ruido de las zonas sensorizadas, lo que permite alertar a las instituciones para que tomen las medidas oportunas, y se puede informar a los ciudadanos de la situación de la contaminación acústica en base a los datos para que puedan realizar protestas ante las instituciones en base a información contrastada.

Para que la información contenida en los datos sea comprensible para cualquiera, se utilizan descripciones lingüísticas generadas automáticamente. La lógica difusa se puede utilizar para el tratamiento de los datos de ruido, encontrando en la literatura trabajos que evalúan el riesgo [1], [2], realizan un análisis del ruido [3], [4], predicen el ruido [5], crean diseños tratando de evitar el ruido [6], etc. Este trabajo presenta un método para generar descripciones lingüísticas diarias a partir de los datos de ruido captados por las estaciones de medición que considera los cuatro periodos diarios para los que se dispone de información (día, tarde, noche y día completo). Para probar el método propuesto, se han utilizado los datos disponibles de la ciudad de Madrid para generar descripciones que permitan analizar la influencia de Covid-19 en la contaminación acústica.

Este trabajo ha sido financiado por el Departamento de Tecnologías y Sistemas de Información de la Universidad de Castilla-La Mancha, por la Escuela de Ingeniería Industrial y Aeroespacial de Toledo y por la Escuela Superior de Informática de Ciudad Real.

El resto del documento está organizado como sigue: la Sección II describe algunos conceptos relevantes en relación con el sonido y el ruido. La Sección III presenta el proceso llevado a cabo para representar y clasificar el ruido utilizando conjuntos difusos. La Sección IV muestra nuestra propuesta para generar descripciones lingüísticas de un día. La Sección V presenta algunas pruebas realizadas con datos de ruido en cinco zonas de Madrid para realizar una primera aproximación a la detección del efecto del encierro domiciliario debido a Covid-19. Finalmente, las conclusiones y los trabajos futuros se detallarán en la Sección VI.

II. Contaminación acústica

Las definiciones de sonido, acústica y ruido [7] que se dan a continuación son necesarias para la correcta comprensión de este trabajo. El sonido se define como la variación de presión producida en un medio (sólido, líquido o gaseoso) por un elemento vibrante que puede ser detectado por el oído humano. La acústica puede definirse como la ciencia que se ocupa de la producción, control, transmisión, recepción y efectos del sonido. El ruido se define como un conjunto de sonidos inarmónicos o desafinados que resultan desagradables para el oído humano, es decir, un sonido molesto.

La medición del ruido está influida por la distancia a la fuente de ruido, que puede ser una fuente puntual, una fuente espacial o una fuente lineal. En las ciudades, la principal fuente de ruido es el tráfico, por ejemplo, en España es la causa del 99% del ruido urbano. El estudio del ruido utiliza las mismas magnitudes que para el sonido, que en su expresión más simple produce la formación de una onda sinusoidal con las siguientes magnitudes: velocidad, longitud de onda, periodo y amplitud.

Dos conceptos importantes en la medición del ruido que a veces se confunden, probablemente porque ambos se miden en decibelios (dB), son el "nivel de potencia sonora" y el "nivel de presión sonora". El nivel de potencia sonora emitido por una fuente sonora determina la cantidad de ruido que produce, mientras que el nivel de presión sonora determina la cantidad de sonido que llega a un punto determinado. El nivel de presión sonora depende de factores como la distancia de la fuente al foco, la dirección o la existencia de otros ruidos en el entorno (ruido de fondo).

Las estaciones de medición de ruido, conocidas como estaciones NMT, suelen utilizar el nivel de presión sonora ponderado A (dBA) para medir el ruido en una zona. En este trabajo hemos utilizado datos de ruido extraídos de la página web del "Portal de datos abiertos del Ayuntamiento de Madrid"[1]. Los conjuntos de datos proporcionan seis mediciones de presión sonora para cada uno de los cuatro periodos de tiempo en los que se puede dividir un día. El primer periodo corresponde al día completo (T) mientras que los otros tres periodos dividen el día en el periodo diurno (D) de 7:00 a 19:00, el periodo vespertino (E) de 19:00 a 23:00 y el periodo nocturno (N) de 23:00 a 7:00. Para cada periodo, la medida L_{Aeq} es el nivel de presión sonora continuo equivalente ponderado A determinado a lo largo de todo el día (periodo T). Además, se proporcionan otras cinco medidas de presión sonora con ponderación de frecuencia A y ponderación de tiempo lento para indicar el nivel que se supera durante un tiempo de observación. Estas medidas son L_{AS01}, L_{AS10}, L_{AS50}, L_{AS90} y L_{AS99} para indicar el ruido superado durante 1%, 10%, 50%, 90% y 99% respectivamente. Los valores de L_{AS01} y L_{AS99} se aproximan al ruido mínimo y máximo alcanzado durante el periodo estudiado respectivamente. En la Tabla I se detalla una muestra correspondiente al 9 de abril de 2020 de la estación situada en la Plaza del Emperador Carlos V de Madrid, donde se muestran los cuatro periodos en las filas y las seis mediciones para cada uno de estos periodos en las columnas.

TABLA I
Muestras de un día en la estación NMT Carlos V de Madrid

Periodo	L_{Aeq}	L_{AS01}	L_{AS10}	L_{AS50}	L_{AS90}	L_{AS99}
D	62.1	69.5	64.8	58.4	52.6	48.7
E	63.5	71.2	66.8	60.2	54.3	50.6
N	59.5	68.0	61.7	54.1	46.4	43.0
T	61.7	69.8	64.8	57.5	50.2	44.0

III. Descripción lingüística del ruido

En esta sección se muestra cómo se pueden representar los datos de ruido para ser utilizados en la generación de descripciones, también se detallan algunos conceptos utilizados por el método presentado para generar las descripciones. En concreto, se utiliza la lógica difusa, ya que permite representar lingüísticamente valores numéricos interrelacionados, como es el caso de las mediciones de presión sonora. Para representar los valores de un periodo se utiliza un conjunto difuso con una función de pertenencia gaussiana (Sección III-A). Para categorizar estos conjuntos difusos, hemos creado un conjunto de etiquetas lingüísticas con una función de pertenencia gaussiana que representa las clases en las que se clasifica el ruido (Sección III-B). Finalmente, para clasificar el ruido de un periodo, se compara el conjunto difuso gaussiano con las etiquetas del conjunto de etiquetas (las clases) seleccionando la clase que ofrece la mayor intersección con el conjunto (Sección III-C).

A. Obtención de un conjunto difuso con función de pertenencia gaussiana a partir de datos con ruido

Las medidas L_{AS01}, L_{AS10}, L_{AS50}, L_{AS90} y L_{AS99} se utilizarán para generar un conjunto difuso que represente una muestra de un periodo. La información obtenida de las estaciones permite conocer la distribución de los valores que superan un umbral durante un 1%, 10%, 50%, 90% y 99% que pueden ser representados por un conjunto difuso con una función de pertenencia gaussiana. La función gaussiana está definida por la Ecuación 1.

$$a * e^{-\frac{(x-b)^2}{2*c^2}} \quad (1)$$

[1]https://cutt.ly/Lj6O0xP

donde a, b y c son constantes reales y $c > 1$. a es el valor máximo que toma la función, b es la posición central de la función y c la desviación típica que rige la amplitud de la función.

En este trabajo se han utilizado los siguientes valores para estos tres parámetros: $a = 1$ (una función de pertenencia da valores de salida en el intervalo $[0, 1]$), $b = \rho$ (media aritmética) y $c = \sigma$ (desviación estándar). Como puede verse, la función gaussiana está determinada por la media y la desviación estándar. Por tanto, para obtener el conjunto difuso que la representa, hay que calcular estos dos valores para L_{AS01}, L_{AS10}, L_{AS50}, L_{AS90} y L_{AS99}. Por ejemplo, para el día completo que se muestra en la última fila de la Tabla I (periodo T) se utiliza como entrada $[69.8, 64.8, 57.5, 50.2, 44.0]$, entonces $\rho = 57.26$ y $\sigma = 9.38$, aplicando la ecuación se obtiene el conjunto difuso mostrado en la Figura 1. Como se puede observar, establece el centro en 57.26 y para $x = 44$ y $x = 69.8$ tiene valores muy bajos.

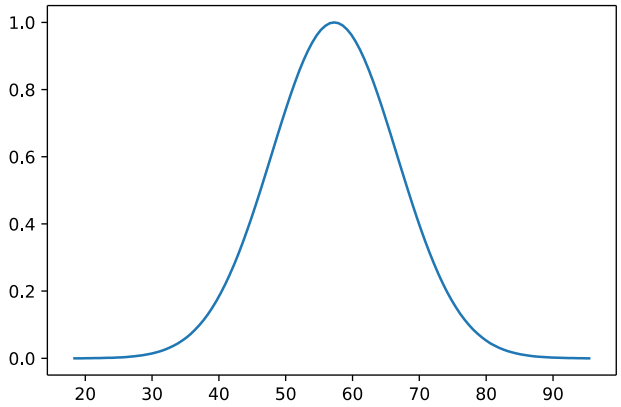


Fig. 1. Conjunto obtenido para el periodo T mostrado en la última fila de la Tabla I.

B. Clasificación de la contaminación ambiental mediante conjuntos difusos

Para clasificar los conjuntos difusos obtenidos en el apartado anterior, se necesita un conjunto de etiquetas lingüísticas que definan las clases a considerar, que se denominarán $CLASSES$. El proceso para obtener las etiquetas que representan las clases es el mismo que el utilizado para calcular los conjuntos difusos. Con la ayuda de un experto, se han definido los valores L_{AS01}, L_{AS10}, L_{AS50}, L_{AS90} y L_{AS99} para cada etiqueta del conjunto que definen la clase (Tabla II), generando posteriormente la etiqueta con función de pertenencia gaussiana mediante el método mostrado anteriormente. Se han seguido las directrices de la OMS, definiendo el "ruido molesto" en un valor cercano a 60 dBA y clasificando el "ruido perjudicial" en 70 dBA o superior, mientras que las otras tres etiquetas dividen el ruido no molesto para el oído humano en tres clases (nulo, imperceptible y aceptable). La Figura 2 muestra gráficamente las etiquetas obtenidas para cada clase. Obsérvese que las etiquetas podrían hacer uso de cualquier otro tipo de función de pertenencia, por ejemplo, trapezoidal, triangular, etc.

TABLA II
VALORES DE L_{AS01}, L_{AS10}, L_{AS50}, L_{AS90} Y L_{AS99} UTILIZADOS PARA DETERMINAR LAS CLASES

Periodo	L_{AS01}	L_{AS10}	L_{AS50}	L_{AS90}	L_{AS99}
nulo	20.0	25.0	30.0	35.0	40.0
imperceptible	30.0	35.0	40.0	45.0	50.0
aceptable	40.0	45.0	50.0	55.0	60.0
molesto	50.0	55.0	60.0	65.0	70.0
perjudicial	60.0	65.0	70.0	75.0	80.0

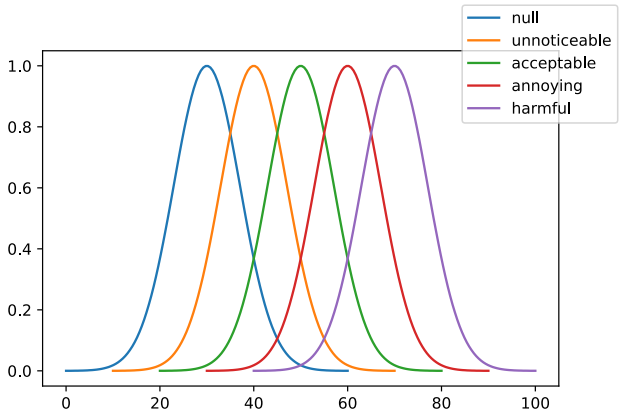


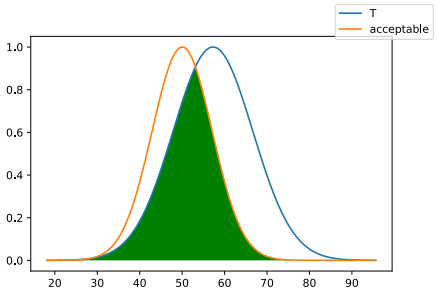
Fig. 2. Conjunto de etiquetas lingüísticas para clasificar el ruido ($CLASSES$).

C. Clasificación de un conjunto difuso mediante el conjunto de etiquetas que clasifica el ruido

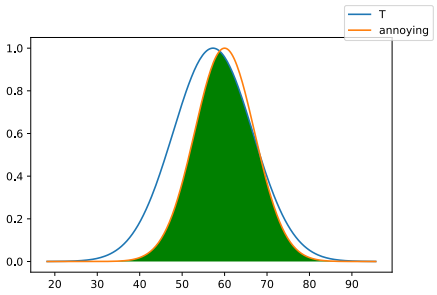
Una vez que la información de entrada se puede representar como un conjunto difuso y se dispone del conjunto $CLASSES$ para su clasificación, queda detallar la forma en que se realiza el proceso de clasificación, que se basa en el concepto de intersección entre conjuntos difusos. La intersección entre dos conjuntos difusos A y B se define como otro conjunto difuso C cuya función de pertenencia es $\mu_C(x) = \mu_A(x) \cap \mu_B(x)$. La Figura 3 muestra varios ejemplos de la intersección de un conjunto difuso y una etiqueta con el conjunto de intersección resultante entre los dos conjuntos resaltados en verde.

Para clasificar un conjunto difuso se realiza la intersección de las etiquetas $CLASSES$ con el conjunto difuso, seleccionando la clase que ofrece la mayor intersección de todas, es decir, se asigna la clase de $CLASSES$ cuya intersección con el conjunto difuso cubre el mayor porcentaje de la función de pertenencia del conjunto difuso a comparar. Por ejemplo, en la Figura 3 se muestra la comparación del conjunto difuso T con las etiquetas de clase *aceptable*, *molesto* y *perjudicial*. Dado que la intersección cubre 58%, 73% y 38% del área

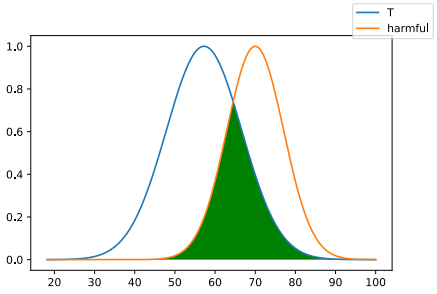
de la función de pertenencia de T para estas tres clases se determina que la clase del conjunto difuso T es *molesto*. De este modo, la clase de un ruido puede obtenerse mediante un conjunto de etiquetas.



(a) y perjudicial T y aceptable (58%)



(b) Intersección entre T y molesto (73%)



(c) Intersección entre T y perjudicial (38%)

Fig. 3. Ejemplos de selección de clases de un conjunto difuso T

IV. Generación de descripciones lingüísticas

A continuación se presenta el método propuesto para generar una descripción de los datos diarios que hace uso de los conceptos mostrados en el apartado anterior. Para generar las descripciones diarias se utilizarán los cuatro periodos diarios (D, E, N, T) y las cinco medidas L_{AS01}, L_{AS10}, L_{AS50}, L_{AS90} y L_{AS99} de cada periodo. El Algoritmo 1 muestra los pasos seguidos para la generación de la descripción. Necesita como entrada un vector que contenga las medidas del día para cada periodo, llamado $DATA$. Las sentencias que componen el algoritmo son las siguientes:

- Línea 1: calcula el conjunto de etiquetas que representan las clases como se muestra en la sección III-B y genera el conjunto de etiquetas $CLASSES$ que contiene las etiquetas de las clases (Figura 2).
- Líneas 2-4: Este bucle obtiene los conjuntos difusos con función de pertenencia gaussiana que representan los datos para cada uno de los períodos de las mediciones del nivel de presión sonora ($DATA$) y almacena el resultado en el vector $GAUSS$. El método para completar este paso se describió en la Sección III-A.
- Líneas 5-8: Este bucle calcula la clase que representa cada uno de los periodos junto con el porcentaje de área que cubre la función de pertenencia del conjunto difuso de entrada por esa clase. Esto se hace realizando la intersección entre cada conjunto difuso en $GAUSS$ con las etiquetas en $CLASSES$. Se selecciona la clase que cubre el mayor porcentaje del área de la función de membresía del conjunto difuso (Sección III-C).
- Línea 6: obtiene la clase asignada al conjunto difuso para ese periodo ($GAUSS[class]$) almacenándola en $LABEL$.
- Línea 7: calcula el porcentaje del conjunto $GAUSS[class]$ que cubre la intersección con la etiqueta seleccionada de $CLASSES$ ($CLASSES[LABEL[period]]$) y lo almacena en el vector $PERC$.
- Línea 9: genera la descripción lingüística utilizando $LABEL$ y $PERC$ para completar la siguiente plantilla:

  En general, fue un día con ruido $LABEL[T]$ durante el $PERC[T]\%$ del tiempo. Durante el día hubo un ruido $LABEL[D]$ que ocupó el $PERC[D]\%$ del tiempo. Por la noche, durante el $PERC[E]\%$ del tiempo, se produjo un ruido $LABEL[E]$. Por último, fue una noche con un ruido

Algorithm 1 Descripción del comportamiento de un día

1: $CLASSES \leftarrow$ ObteniendoClases() {Sección III-B. }
{El siguiente bucle calcula las gaussianas (Sección III-A).}
2: **for** $period(p)$ **in** $[T, D, E, N]$ **do**
3: $GAUSS[p] \leftarrow$CalcularGaussiana($DATA[p]$) {$DATA$ es un vector que contiene los datos de entrada. $GAUSS$ es un vector que contiene las gaussianas de cada periodo.}
4: **end for**
{El siguiente bucle calcula los vectores $LABEL$ y $PERC$.}
5: **for** $period(p)$ **in** $[T, D, E, N]$ **do**
6: $LABEL[p] \leftarrow$ SelecClass($GAUSS[p]$, $CLASSES$) {Sección III-C}
7: $PERC[LABEL[p]] \leftarrow$ Intersección($GAUSS[p]$, $CLASSES[LABEL[p]]$)
8: **end for**
{$description$ es la descripción generada utilizando $MAXIMA$.}
9: $description \leftarrow$ GenerarPlantilla($MAXIMA$)

$LABEL[N]$ durante $PERC[N]\%$ de su tiempo

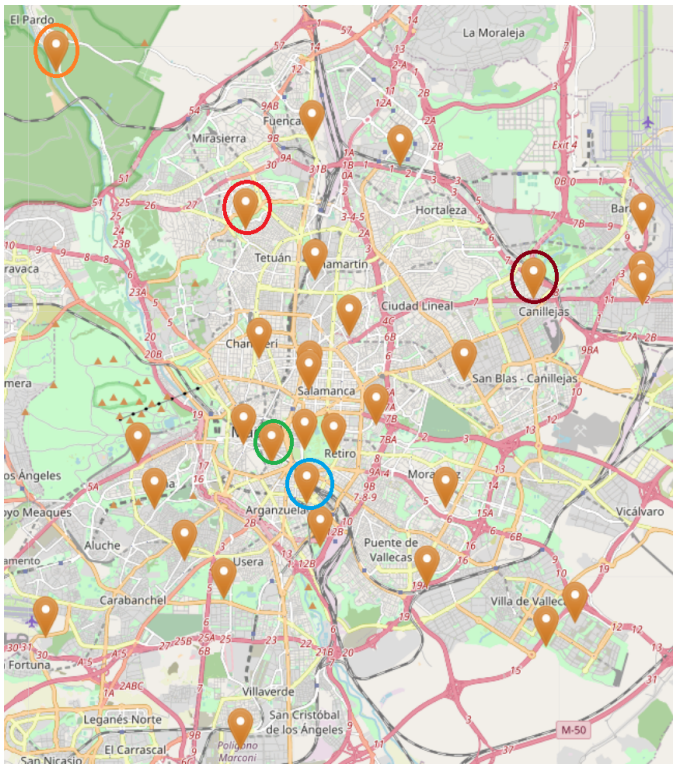


Fig. 4. Ubicación de las zonas de medición utilizadas en las pruebas.

V. Tests

Para probar el método presentado se han generado descripciones para cinco áreas de Madrid situadas en zonas con diferentes características identificadas en la Figura 4. Situadas en el centro de Madrid se encuentran las estaciones de Plaza del Carmen (círculo verde) y Carlos V (círculo azul), que son zonas con bastante vida urbana y tráfico, especialmente la Plaza del Emperador Carlos V, cercana al nudo de comunicaciones de Atocha. Algo alejadas del centro están las estaciones de El Barrio del Pilar (círculo rojo), situada en la Avenida de Monforte Lemos junto al Parque de la Vaguada, y Campo de las Naciones (círculo granate), con la estación situada en la rotonda del monumento a Don Juan de Borbón, que tiene un tráfico constante. Lejos del centro está la estación de El Pardo (círculo naranja), un barrio situado en un bosque, una zona muy tranquila. Se han elegido dos miércoles para generar las descripciones, concretamente el 5 de febrero de 2020 y el 1 de abril de 2020. El día 5-2-2020 corresponde a una fecha anterior a la del encierro domiciliario de España por Covid-19 (15 de marzo de 2020), es decir, representa la actividad normal, mientras que el día 1-4-2020 es un día de encierro domiciliario. De este modo, las descripciones nos permitirán evaluar el efecto del confinamiento domiciliario en la contaminación acústica, aunque en el futuro se deberá realizar un estudio más amplio para poder extraer conclusiones válidas. En el apartado de bibliografía, existen algunos estudios sobre el efecto del Covid-19 en la contaminación acústica en grandes ciudades, como Dublín [8], Roma [9] o Madrid [10].

TABLA III
Descripción lingüística de la estación de medida Carlos V

Carlos 5 (1-02-2020)
En general, fue un día con ruido perjudicial durante el 71,0% del tiempo. Durante el día hubo ruido perjudicial el 98,0% del tiempo. Por la tarde, durante el 99,0% del tiempo, se alcanzó un ruido perjudicial. Por último, por la noche hubo ruido molesto durante el 84,0% del tiempo.
Carlos 5 (5-04-2020) - confinamiendo domiciliario
En general, fue un día con ruido molesto durante el 80,0% del tiempo. Durante el día hubo ruidos molestos que ocuparon el 84,0% del tiempo. Por la tarde, durante el 94,0% del tiempo, se alcanzó un ruido molesto. Por último, por la noche hubo ruido molesto durante el 75,0% del tiempo.

Tabla III muestra las descripciones obtenidas para la estación situada en Carlos V. Como se puede observar, se trata de una zona muy ruidosa en día laborable, obteniendo una clasificación de "ruido perjudicial" durante todos los periodos y persistiendo durante la mayor parte del tiempo. Se observa que durante el encierro recibe una clasificación de ruido menor, aunque se siguen manteniendo valores superiores a las recomendaciones de la OMS durante casi todos los periodos completos. Esto puede deberse a la concentración de tráfico continuo e intenso en la zona.

TABLA IV
Descripción lingüística de la estación de medida de la Plaza del Carmen

Plaza del Carmen (1-02-2020)
En general, fue un día con ruido molesto durante el 76,0% del tiempo. Durante el día hubo ruido molesto el 91,0% del tiempo. Por la tarde, durante el 95,0% del tiempo hubo ruido molesto. Por último, por la noche hubo ruido molesto durante el 68,0% del tiempo.
Plaza del Carmen (5-04-2020) - confinamiento domiciliario
En general, fue un día con un ruido aceptable durante el 73,0% del tiempo. Durante el día hubo un ruido aceptable durante el 60,0% del tiempo. Por la tarde, durante el 75,0% del tiempo, se obtuvo un ruido aceptable. Por último, por la noche hubo un ruido aceptable durante el 76,0% del tiempo.

En la Tabla IV se detallan las descripciones obtenidas para la estación situada en la Plaza del Carmen. Durante un día laborable es una zona con ruido molesto durante todo el día, aunque se observa que por la noche el ruido molesto se mantiene durante menos tiempo que el resto del día. El confinamiento causó un efecto positivo en esta zona obteniendo un ruido aceptable durante más del 60% del tiempo de todos los periodos.

La Tabla V muestra las descripciones de la estación situada en el Barrio del Pilar. En un día laborable es un barrio ruidoso todo el tiempo en los periodos diurnos y vespertinos, pero por la noche se consigue un ruido aceptable. En confinamiento la calidad de vida mejoró con respecto a la contaminación acústica, convirtiéndose en un barrio con un ruido aceptable durante todo el día. En este caso, esto se debe al fuerte des-

TABLA V
DESCRIPCIÓN LINGÜÍSTICA DE LA ESTACIÓN DE MEDIDA DEL BARRIO PILAR

Barrio Pilar (1-02-2020)
En general, fue un día con ruido molesto durante el 66,0% del tiempo. Durante el día hubo ruido molesto el 100,0% del tiempo. Por la tarde, durante el 100,0% del tiempo, hubo ruido molesto. Por último, durante la noche hubo un ruido aceptable durante el 93,0% del tiempo.
Barrio Pilar (5-04-2020) - confinamiento domiciliario
En general, fue un día con un ruido aceptable durante el 63,0% del tiempo. Por la mañana hubo un ruido aceptable durante el 74,0% del tiempo. Por la tarde, el ruido fue aceptable el 67,0% del tiempo. Por último, fue una noche con un ruido aceptable durante el 66,0% de su tiempo.

censo del tráfico en la zona provocado por el confinamiento, ya que los residentes no pueden circular libremente.

TABLA VI
DESCRIPCIÓN LINGÜÍSTICA DE LA ESTACIÓN DE MEDIDA DE CAMPO DE LAS NACIONES

Campo de las Naciones (1-02-2020)
En general, fue un día con un ruido aceptable durante el 63,0% del tiempo. Durante el día hubo un ruido molesto que ocupó el 100,0% del tiempo. Por la tarde, durante el 91,0% del tiempo, hubo ruido molesto. Por último, por la noche hubo un ruido aceptable durante el 70,0% del tiempo.
Campo de las Naciones (5-04-2020) - confinamiento domiciliario
En general, fue un día con un ruido aceptable durante el 76,0% del tiempo. Durante el día hubo un ruido aceptable durante el 76,0% del tiempo. Por la tarde, durante el 77,0% del tiempo, se obtuvo un ruido aceptable. Por último, por la noche hubo un ruido aceptable durante el 80,0% del tiempo.

La estación de Campo de las Naciones está situada en una rotonda con tráfico, lo que provoca un nivel de ruido molesto durante el día y la tarde, con un nivel de ruido aceptable por la noche (Tabla VI). Cuando se produjo el confinamiento, se alcanzó un nivel de ruido aceptable durante más del 75% de las veces.

TABLA VII
DESCRIPCIÓN LINGÜÍSTICA DE LA ESTACIÓN DE MEDICIÓN DE EL PARDO

El Pardo (1-02-2020)
En general, fue un día con un ruido aceptable durante el 54,0% del tiempo. Durante el día hubo un ruido molesto que ocupó el 76,0% del tiempo. Por la tarde, durante el 72,0% del tiempo, hubo un ruido aceptable. Por último, por la noche hubo un ruido imperceptible durante el 58,0% del tiempo.
El Pardo (5-04-2020) - confinamiento domiciliario
En general, fue un día con un ruido aceptable durante el 56,0% de su tiempo. Durante el día hubo un ruido aceptable durante el 64,0% del tiempo. Por la tarde, durante el 63,0% del tiempo, se obtuvo un ruido aceptable. Por último, por la noche hubo un ruido imperceptible durante el 64,0% del tiempo.

Por último, se ha comprobado que en general El Pardo es una zona con un nivel de ruido aceptable, destacando que el confinamiento ha modificado a mejor los niveles de ruido de la mañana, manteniendo el descanso. Se puede concluir que es una zona sin mucho ruido de forma habitual, por lo que el confinamiento no ha tenido tanto impacto como en las otras zonas estudiadas.

VI. CONCLUSIONES

En este trabajo se ha presentado un método para generar descripciones lingüísticas de la contaminación acústica, más concretamente, se ha detallado un método para generar descripciones de un día que considera cuatro periodos diarios (día, tarde, noche y día completo). Para probar el método propuesto, se han utilizado los datos disponibles de la ciudad de Madrid, seleccionando cinco zonas con diferentes características y considerando dos días, uno con actividad normal y otro con confinamiento domiciliario debido a la Covid-19. De las descripciones obtenidas se puede concluir que el confinamiento ha reducido la contaminación acústica, excepto en un lugar donde ya era aceptable (El Pardo). También se ha comprobado que estas descripciones son válidas para el estudio de la contaminación acústica, ya que son más fáciles de analizar que los datos brutos.

Como trabajo futuro estamos considerando el uso de otros tipos de funciones de pertenencia, tanto para el conjunto de etiquetas lingüísticas como para los conjuntos difusos utilizados para representar los datos tomados de las estaciones. También pretendemos investigar la descripción de intervalos de tiempo más largos, como semanas, meses o años.

REFERENCES

[1] R. Golmohammadi, M. Eshaghi, and M. Reyahi Khoram, “Fuzzy Logic Method for Assessment of Noise Exposure Risk in an Industrial Workplace”, Int J Occup Hyg, vol. 3, no. 2, pp. 49-55, 2011.

[2] S.K. De, B.K. Swain, S. Goswami, and M. Das, “Adaptive noise risk modelling: fuzzy logic approach”, Systems Science & Control Engineering, vol. 5, no. 1, pp. 129-141, 2017.

[3] M. Ghaderi, H. Javadikia, L. Naderloo, M. Mostafaei, and H. Rabbani, “Analysis of noise pollution emitted by stationary MF285 tractor using different mixtures of biodiesel, bioethanol, and diesel through artificial intelligence”, Environmental Science and Pollution Research, vol. 26, pp. 21682–21692, 2019.

[4] M. F. I. Mohd Idris, N. A. Zainol Abidin, and K. A. Abd Aziz, “The Determination of Factors that Influence the Noise Pollution in Malaysia using Fuzzy Logic”, Journal of Computing Research and Innovation, vol. 5, no. 2, pp. 1-10, Oct. 2020.

[5] V, Nouraniac, H. Gökçekuş, and I.K Umar, “Artificial intelligence based ensemble model for prediction of vehicular traffic noise”, Environmental Research, vol. 10, 108852, 2020.

[6] P.T. Nguyen, “Construction site layout planning and safety management using fuzzy-based bee colony optimization model”. Neural Comput & Applic, 2020 (https://doi.org/10.1007/s00521-020-05361-0).

[7] R J Peters, B.J. Smith, and M. Hollins, “Acoustics and Noise Control”, Third Edition, 2011.

[8] B. Basu, E. Murphy, A. Molter, A. Sarkar Basu, S. Sannigrahi, M. Belmonte, and F. Pilla, “Investigating changes in noise pollution due to the COVID-19 lockdown: The case of Dublin, Ireland”, Sustainable Cities and Society, vol. 65, 102597, February 2021.

[9] F. Aletta, S. Brinchi, S. Carrese, A. Gemma, C. Guattari, L. Mannini, and S.M. Patella, “Analysing urban traffic volumes and mapping noise emissions in Rome (Italy) in the context of containment measures for the COVID-19 disease”, Noise Mapping, vol. 7, pp. 114-122, 2020.

[10] C. Asensio, I. Pavón, and G. de Arcas, “Changes in noise levels in the city of Madrid during COVID-19 lockdown in 2020”, The Journal of the Acoustical Society of America, vol. 148, pp. 1748, 2020.

Un sistema para la descripción automática en lenguaje natural de gráficas de sectores: aplicación en datos de calidad del aire

Andrea Cascallar-Fuentes*, Javier Gallego-Fernández*, Alejandro Ramos-Soto*, Anthony Saunders†, Alberto Bugarín-Diz*

*Centro Singular de Investigación en Tecnoloxías Intelixentes (CiTIUS), Universidade de Santiago de Compostela, Spain
javier.gallego.fernandez@rai.usc.es, {andrea.cascallar.fuentes, alejandro.ramos, alberto.bugarin.diz}@usc.es

†MeteoGalicia, Xunta de Galicia, Santiago de Compostela, Spain
{calidadedoaire.cma}@xunta.es

***Resumen*—En este trabajo presentamos un modelo basado en la generación de lenguaje natural y en la lógica borrosa para la generación automática de descripciones lingüísticas a partir de datos numéricos y su aplicación real en el ámbito de la información ambiental. Basado en dicho modelo, describimos el sistema ICA2Text, que genera automáticamente descripciones en lenguaje natural sobre el índice de calidad del aire (ICA), que es un indicador estándar utilizado por todas las agencias meteorológicas a nivel mundial. ICA2Text es una aplicación real que opera sobre los datos del ICA proporcionados por la Red de Calidad del Aire de la Agencia de Meteorología de Galicia (MeteoGalicia). Siguiendo la metodología estándar de evaluación en el campo de la generación de lenguaje natural, presentamos los resultados de evaluación manual del sistema por parte de tres expertos meteorólogos. Los resultados de dicha evaluación fueron muy satisfactorios, confirmando empíricamente que las descripciones en lenguaje natural que se generan a partir de los datos resultaron muy adecuadas, tanto en su contenido como en su calidad lingüística. Por ello, el sistema estará operativo en breve como servicio público para los usuarios de la web de MeteoGalicia.**

***Index Terms*—descripciones lingüísticas de datos, sistemas data-to-text, generación de lenguaje natural.**

I. Introducción

Obtener información relevante a partir de grandes cantidades de datos plantea varios retos que no pueden abordarse únicamente con las técnicas tradicionales basadas en la estadística y las visualizaciones gráficas. En general, estos enfoques son muy útiles para obtener información básica de los datos, pero para que los usuarios comprendan la información realmente relevante que hay detrás de los datos, es necesario emplear técnicas que se adapten mejor a las necesidades específicas de cada dominio y que puedan escalar a medida que aumenta la cantidad de datos. En este sentido, la Inteligencia Artificial proporciona a los usuarios herramientas de análisis de conjuntos de datos para extraer información útil, así como herramientas de procesamiento del lenguaje que permiten una comunicación más fluida entre humanos y máquinas.

Una técnica prometedora para este propósito es la Generación de Lenguaje Natural (NLG, por sus siglas en inglés), que permite generar texto a partir de varias fuentes de datos (principalmente numéricos y textuales).

Dentro de este campo, la arquitectura [1] más empleada en la literatura es la siguiente (Figura 1):

- Determinación de contenido: esta fase se centra en decidir qué información será comunicada en el texto.
- Planificación del discurso: en esta etapa se decide el conjunto de mensajes que serán verbalizados y se les asigna un orden y una estructura.
- Planificación de la sentencia: en esta fase se agrupan los mensajes según sea necesario y se eligen las palabras y expresiones que deben ser utilizadas.
- Realización lingüística: en esta etapa se lleva a cabo el proceso de generar el texto resultante, que debe ser morfológica y ortográficamente correcto.

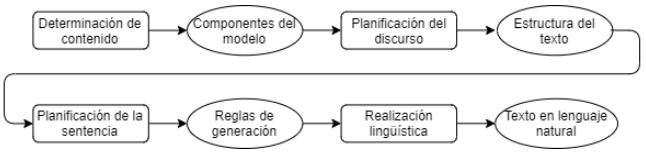


Figura 1. Arquitectura NLG propuesta en [1]. Los rectángulos muestran las fases más importantes mientras que las elipses muestran la salida de cada una de ellas.

Dentro del NLG, los sistemas data-to-text (D2T) [2] generan automáticamente textos a partir de grandes conjuntos de datos numéricos o simbólicos, proporcionando una información comprensible que no podría producirse de otro modo. Los sistemas D2T incluyen: *i)* una etapa de análisis de datos en la que se extrae la información relevante de los mismos y *ii)* una etapa de generación en la que se transmite la información en lenguaje natural.

Además, también relacionado con el NLG, desde el campo de la lógica difusa se propusieron varios enfoques para generar

descripciones lingüísticas de datos (LDD, por sus siglas en inglés), basadas en el uso de términos lingüísticos modelados como conjuntos difusos. Siguiendo este enfoque, algunas aproximaciones [3] resumen en una forma lingüística una o más variables numéricas y sus valores, utilizando la noción general de protoforma [4] y de forma más específica, sentencias cuantificadas borrosas que pueden seguir varios tipos de estructura (por ejemplo, "En algunos puntos la temperatura es alta").

Las LDD carecen, en general, de la expresividad de los textos reales, pero no por ello dejan de ser elementos de información útiles que pueden utilizarse como entrada de alto nivel para los sistemas NLG en general y D2T en particular [5], [6]. Se han propuesto aplicaciones de descripciones lingüísticas en diversos campos como en el de la salud [7], negocios [8], ahorro de energía [9] o meteorología [10].

En este trabajo proponemos el sistema ICA2Text, que tiene como objetivo la generación de descripciones lingüísticas a partir de datos sobre el Índice de Calidad del Aire (ICA), que es un indicador estándar de la calidad del aire utilizado por todas las agencias meteorológicas y medioambientales del mundo [11]. Nuestro sistema genera descripciones en lenguaje natural a partir de datos de distribución del ICA en las estaciones de la red de observación meteorológica proporcionadas por la Agencia Meteorológica de Galicia, MeteoGalicia [12].

En el campo de la meteorología existen varios enfoques a lo largo de los años para generar descripciones a partir de los datos de calidad del aire. [13] es un prototipo del sistema TEMSIS centrado en describir si se han superado o no los umbrales (diarios, mensuales...) de alerta temprana para cada contaminante. También el sistema MARQUIS [14] de series temporales de índice de calidad del aire numéricas incluye referencias temporales simples (horas o intervalos específicos) para describir el último valor del ICA y los contaminantes (concentración, información de archivo y previsión). En [15] se propone una solución centrada en la generación de descripciones del índice de calidad del aire para una ventana temporal que incluye tres valores diarios.

Este trabajo está estructurado de la siguiente manera: en primer lugar, en la sección II presentamos el contexto del problema gestionado en este trabajo y la arquitectura de nuestro enfoque. En la Sección III describimos en profundidad los detalles del modelo para la descripción en lenguaje natural de la distribución del Índice de Calidad del Aire. En la sección IV se presenta la validación por expertos humanos y sus resultados. Por último, en la sección V ofrecemos algunas observaciones finales y una discusión sobre trabajo futuro.

II. Contexto del problema

La presencia de contaminantes en el aire y, por lo tanto, el deterioro de la calidad del aire, pueden tener efectos nocivos para la salud de las personas.

El Índice de Calidad del Aire (ICA) es una variable simbólica del tiempo que representa la calidad del aire en cada momento midiendo la presencia y densidad de diversos tipos de partículas contaminantes en la atmósfera, utilizada por las agencias meteorológicas y los gobiernos para informar a la población sobre la calidad del aire.

En concreto, los datos con los que trabajamos describen el Índice de Calidad del Aire en la red de 50 estaciones meteorológicas (Figura 2) que envían datos actualizados cada hora en tiempo real en Galicia. Se trata de datos oficiales proporcionados por Meteogalicia, que es la agencia de meteorología oficial de la Xunta de Galicia [12].

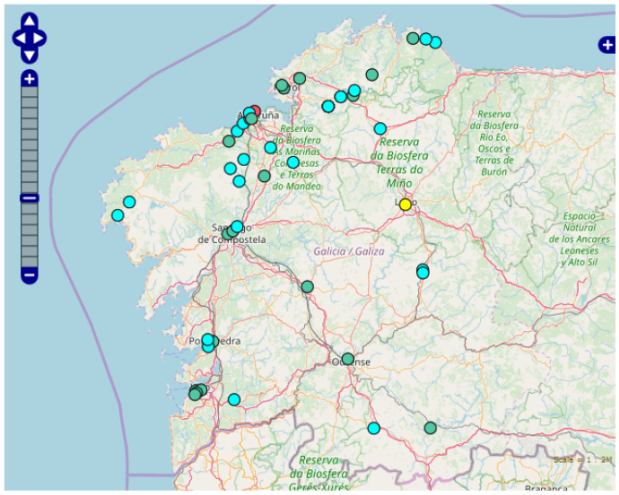


Figura 2. Mapa de las estaciones meteorológicas de MeteoGalicia.

Este servicio se ha actualizado recientemente debido a un proceso de unificación del Índice de Calidad del Aire por parte de diferentes agencias meteorológicas, basado en los criterios de la Agencia Europea de Medio Ambiente [11]. Actualmente, cuenta con seis etiquetas con una percepción positiva, neutra o negativa (Tabla I).

Estas etiquetas están representadas por un código de colores en el que, por ejemplo, el color morado significa "pésima" mientras que el amarillo significa "buena".

Para determinar el valor de calidad del aire adecuado para una situación, este servicio mide cinco contaminantes diferentes: dióxido de azufre (SO_2), dióxido de nitrógeno (NO_2), partículas en suspensión con un diámetro menor o igual a 2.5 micras ($PM25$), partículas en suspensión con un diámetro entre 2.5 y 10 micras ($PM10$) y ozono (O_3).

A partir de los datos obtenidos de las estaciones meteorológicas, MeteoGalicia proporciona representaciones gráficas de la distribución de cada valor de calidad del aire en tiempo real para todas las estaciones gallegas. MeteoGalicia representa estos valores a través de un gráfico de sectores que incluye una leyenda con la lista de estaciones con índice de calidad del aire "malo" o "muy malo" junto con el contaminante causante de esa situación. En la Figura 3 se muestra un ejemplo de la distribución de las etiquetas del ICA, donde el 58 % de las estaciones tiene una calidad del aire "muy buena" y el 38 % de las estaciones tienen una calidad "buena". Por otro lado, una estación tiene una calidad del aire "moderada" y

Tabla I
ETIQUETAS DEL ÍNDICE DE CALIDAD DEL AIRE CON SUS PERCEPCIONES E ÍNDICES NUMÉRICOS ASOCIADOS.

Percepción	Positivo		Neutro	Negativo		
Etiqueta	Buena	Favorable	Regular	Mala	Muy mala	Pésima
Índice	0	1	2	3	4	5

también una única estación tiene una situación "mala" debido al contaminante PM10.

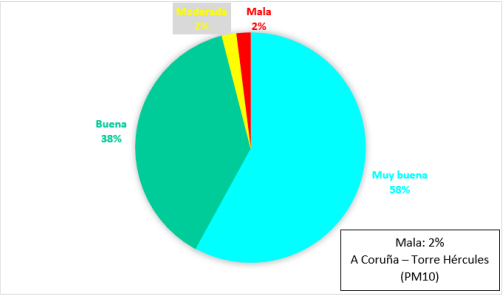


Figura 3. Distribución en tiempo real de los valores de calidad del aire.

III. DESCRIPCIONES LINGÜÍSTICAS DE LA DISTRIBUCIÓN DEL ÍNDICE DE CALIDAD DEL AIRE

Los cuantificadores lingüísticos en el lenguaje son una herramienta muy potente para representar y describir el conocimiento sobre la cantidad de elementos que cumplen determinadas propiedades, cuyo número, al menos desde el punto de vista pragmático, suele encontrarse entre una (cuantificadores unarios) y cuatro (cuantificadores cuaternarios). Las sentencias cuantificadas unarias tienen la siguiente estructura: "*Q X* son *S*" donde *Q* es un cuantificador (por ejemplo, "la mayoría"), *X* es un conjunto referencial (por ejemplo, "días"), y *S* es un valor lingüístico (por ejemplo, "lluvia"). Así, un ejemplo de enunciado cuantificado unario es "La mayoría de los días son lluviosos".

En el caso de MeteoGalicia, se recogen en tiempo real datos de valores del índice de calidad del aire (ICA) de las 50 estaciones meteorológicas que componen su la Red [16]. Debido a la importancia que tiene esta información ya que puede afectar a la salud de las personas, surge el interés en proporcionar descripciones en lenguaje natural sobre los datos del ICA, que normalmente se presentan de forma gráfica. Por lo tanto, hemos definido en colaboración con los expertos de MeteoGalicia los requisitos de las descripciones lingüísticas que luego se proyectaron en las diferentes etapas de la arquitectura NLG.

III-A. Determinación de contenido

A partir de los datos del ICA de los que disponemos, surge el interés de generar descripciones en cuanto a la distribución de sus etiquetas a lo largo de la red de estaciones utilizando los porcentajes. Sin embargo, generar descripciones incluyendo porcentajes puede no ser atractivo para los usuarios, por lo tanto, para describir la distribución de las etiquetas del índice de

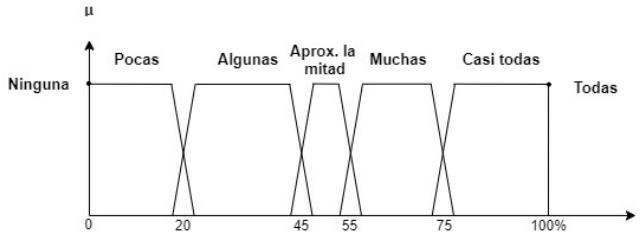


Figura 4. Representación gráfica de la definición de los cuantificadores para la descripción de la distribución del índice de calidad del aire (porcentaje de estaciones en la red meteorológica).

calidad del aire, utilizamos la partición de siete cuantificadores "Ninguna", "Pocas", "Algunas", "Aproximadamente la mitad", "Muchas", "Casi todas", "Todas" representados en la Figura 4, lo que permite verbalizar de forma imprecisa el porcentaje de estaciones que cumplen una etiqueta determinada. Esto resulta más amigable para los usuarios que un porcentaje, puesto que para este sistema no se demanda una descripción precisa.

Siendo *Q* los cuantificadores previamente definidos, *X* las estaciones de la red y *S* las etiquetas del ICA, generamos descripciones unarias, como, por ejemplo, "Muchas estaciones tienen una calidad del aire mala".

A modo de ejemplo, partiendo de un conjunto de datos formado por la dupla ID de estación y valor del ICA para las 50 estaciones (p.e., 2, 3, 52, 1, 85, 1, 107, 2, 164, 2, ...), en esta fase se determinan las piezas de información que se van a incluir en la descripción textual:

- Muchas estaciones tienen calidad del aire muy mala.
- Muchas estaciones tienen calidad del aire mala.
- Rodís tiene calidad pésima debido al NO2.
- Marraxón tiene calidad pésima debido a SO2.
- Pastoriza tiene calidad pésima debido a PM10.
- Magdalena tiene calidad pésima debido a O3.
- Río Cobo tiene calidad muy buena.
- Laza tiene calidad muy buena.
- Mourence tiene calidad muy buena.

III-B. Planificación del documento

Las descripciones en lenguaje natural están formadas por las siguientes partes:

- Descripción general que resume globalmente la situación, con una proposición cuantificada como "Muchas de las estaciones tienen una calidad del aire muy mala o mala".
- Intensificación, que destaca las estaciones que presentan valores de calidad del aire que cumplan en mayor medida con la misma percepción que la descripción general. Esta

parte solo se incluye si hay alguna estación que cumpla esta condición. Por ejemplo, "Destaca Rodís, con una calidad del aire pésima debido al NO2".
- Excepción, que destaca las estaciones que tienen valores de calidad del aire cuya percepción es contraria a la de la descripción general. Esta parte solo se incluye si existe alguna estación con un valor destacable. Por ejemplo, "Por el contrario, Laza tiene una calidad muy buena".

En el ejemplo anterior:
- Muchas estaciones tienen calidad del aire muy mala o mala.
- Rodís tiene calidad pésima debido al NO2.
- Marraxón tiene calidad pésima debido a SO2.
- Pastoriza tiene calidad pésima debido a PM10.
- Magdalena tiene calidad pésima debido a O3.
- Río Cobo tiene calidad muy buena.
- Laza tiene calidad muy buena.
- Mourence tiene calidad muy buena.

III-C. Planificación de la sentencia

Estas reglas de planificación están basadas en las máximas de Grice [17], que pueden resumirse como sigue:
- Calidad: las normas deben estar descritas con precisión.
- Cantidad: las reglas deben contener suficiente información para ser comprensibles sin proporcionar más información de la necesaria.
- Relevancia: las reglas deben contener las reglas importantes para este modelo.
- Forma: las reglas deben estar claramente definidas evitando descripciones ambiguas y deben estar ordenadas en función de la estructura de la descripción lingüística.

Las reglas para la descripción general consisten en:
- Resaltar si hay un valor del ICA mayoritario cuando cubre un porcentaje de estaciones superior a un umbral definido, por ejemplo "Muchas de las estaciones tienen una calidad del aire muy mala" [1].
- Destacar si hay dos etiquetas del ICA con la misma percepción que cubren un porcentaje de estaciones por encima de un umbral fijo, por ejemplo "'Muchas de las estaciones tienen una calidad del aire muy mala".
- Si no hay un ICA predominante, se resaltan los peores valores, por ejemplo "Ourense tiene una calidad del aire extremadamente mala debido al contaminante PM10".

Para la intensificación y la excepción, la descripción debe seguir estas reglas:
- Una vez que se ha descrito el ICA predominante en la parte general de la descripción, el resto de valores se consideran intensificaciones si son etiquetas con la misma percepción o excepciones en caso contrario. Además, los valores con percepciones negativas incluyen los contaminantes que las producen. Por ejemplo "Todas las estaciones tienen una calidad del aire buena, especialmente Ourense con una calidad del aire muy buena. Por el contrario, Paiosaco tiene una calidad del aire muy mala debido al contaminante O_3".
- Si dos o más estaciones tienen el mismo ICA y causante se agrupan, por ejemplo "Laza y Santiago-Campus tienen una calidad del aire muy mala debido al contaminante PM10".

Siguiendo el ejemplo, en esta fase obtenemos:
- Muchas estaciones tienen calidad del aire muy mala o mala.
- Rodís tiene calidad pésima debido al NO2.
- Marraxón tiene calidad pésima debido a SO2.
- Pastoriza tiene calidad pésima debido a PM10.
- Magdalena tiene calidad pésima debido a O3.
- Río Cobo, Laza y Mourence tienen calidad muy buena.

III-D. Realización

En una descripción es tan importante el contenido como la forma de presentarlo, por lo que, con el objetivo de facilitar su lectura, se incluyen reglas de realización relacionadas con la estructura tanto para los casos de intensificación como para los de excepción: si el número de casos destacados es superior a 2, se dispondrán como una lista, en caso contrario se incluirán ambos como texto plano.

Debido a que la web de MeteoGalicia ofrece información tanto en castellano como en gallego, hemos elaborado las descripciones en ambos idiomas utilizando SimpleNLG-ES [18] y SimpleNLG-GL [19], que son versiones extendidas de SimpleNLG [20] para el castellano y el gallego, respectivamente.

La descripción textual completa resultante en el ejemplo anterior es:

Muchas estaciones tienen calidad del aire muy mala o mala. Destacan 4 estaciones:
- *Rodís tiene calidad pésima debido al NO2.*
- *Marraxón tiene calidad pésima debido a SO2.*
- *Pastoriza tiene calidad pésima debido a PM10.*
- *Magdalena tiene calidad pésima debido a O3.*

Por el contrario, Rio Cobo, Laza y Mourence tienen una calidad muy buena.

IV. Validación

Para la validación de los sistemas de Generación de Lenguaje Natural se ha definido una metodología estándar que considera diferentes dimensiones [21]. En cuanto al objeto de la validación, tenemos: *i*) la validación intrínseca, que mide el rendimiento de un sistema en términos de su efectividad con respecto a los usuarios; y *ii*) la validación extrínseca, enfocada en medir la efectividad de un sistema a la hora de conseguir un determinado fin u objetivo. En cuanto a la forma de realizar la validación, tenemos: *i*) la validación manual, realizada a cargo de evaluadores humanos, que pueden ser tanto expertos como no expertos, en función del objetivo final del sistema, y *ii*) la validación automática, utilizando métricas de rendimiento, que son de uso unánime en sistemas de generación neuronal profundo.

[1] El sistema ICA2Text genera descripciones en lenguaje natural bilingües en español y gallego, pero en este trabajo mostramos los ejemplos solo en español para facilitar la lectura.

Teniendo en cuenta estas consideraciones, y dadas las características del sistema ICA2Text optamos por una validación intrínseca manual realizada por expertos [22]. Así, tres meteorólogos expertos de la Red de Calidad del Aire de la Agencia de Meteorología de Galicia [12] (MeteoGalicia) participaron en la evaluación de la calidad de los descripciones en lenguaje natural en este ámbito y su adecuación para la descripción de la distribución del ICA. En la validación se les presentó al grupo de tres expertos un cuestionario compuesto por 25 ejemplos de casos reales de datos proporcionados por 20 estaciones meteorológicas diferentes, que evaluaron de forma conjunta, consensuando una única respuesta para cada ítem de cada caso. En la Figura 5 mostramos uno de los casos, donde se da una situación de ICA mala. Cada caso está compuesto por un diagrama de sectores y la descripción textual correspondiente generada por ICA2Text, acompañado de cinco ítems (mostradas en la Tabla II), que debían responder utilizando una escala Likert [23] de 5 puntos en el rango [1, 5] donde 1 indica acuerdo total del grupo de expertos con el ítem del cuestionario y 5 indica desacuerdo total. Los ítems hacen referencia a dos dimensiones principales: contenido de la descripción (Q1, Q2) y su forma (Q3, Q4, Q5). El equipo de expertos consensuó en todos los casos la evaluación numérica conjuntamente de la correspondencia entre las descripciones en lenguaje natural y la información proporcionada por el gráfico.

Tabla II
ÍTEMS QUE COMPONEN EL CUESTIONARIO PARA LA EVALUACIÓN POR PARTE DE LOS EXPERTOS.

Código	Cuestión
Q1	Indicar el grado de concordancia entre la descripción lingüística proporcionado y los datos representados en la figura
Q2	Indicar el grado de concordancia entre la descripción lingüística proporcionado y cómo describirías los datos
Q3	Indicar el grado de conformidad con el uso correcto del vocabulario
Q4	Indicar el grado de acuerdo con la organización de la descripción lingüística para facilitar su comprensión
Q5	Indique el grado de acuerdo con la ortografía, la puntuación y la estructura

Ninguno de estos expertos había participado en la definición del sistema ICA2Text ni de su modelo subyacente, puesto que todos los requisitos fueron definidos con la ayuda de un cuarto experto diferente, perteneciente también a la Red de Calidad del Aire de MeteoGalicia. Por lo tanto, se realizó una evaluación totalmente ciega, puesto que no se proporcionaron detalles de ningún tipo acerca de cómo se generaron las descripciones en lenguaje natural.

En la Tabla III se presenta un resumen de los resultados tras la evaluación por parte de los expertos de las dimensiones de contenido (Q1, Q2) y diseño (Q3, Q4, Q5) y el resultado global.

Los resultados muestran que los expertos están de acuerdo con las descripciones en lenguaje natural generadas, ya que la media de las puntuaciones es de 4.80 con una desviación típica de 0.51. Además, en todos los ítems la moda la máxima puntuación posible.

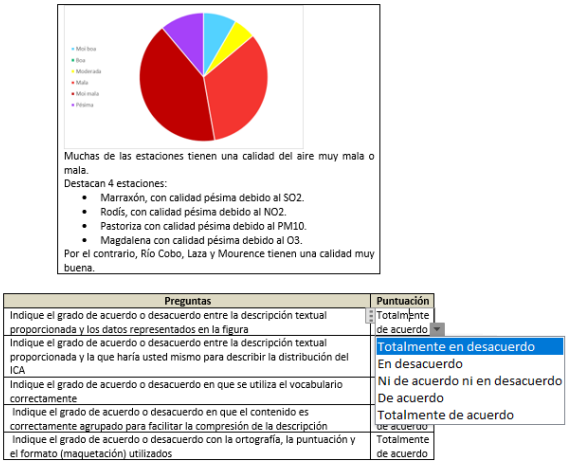


Figura 5. Ejemplo extraído del cuestionario diseñado para la validación de expertos.

Tabla III
RESULTADOS DE LA EVALUACIÓN POR EXPERTOS.

	Media	Desviación típica	Moda
Q1	4.76	0.52	5
Q2	4.68	0.69	5
Q3	4.88	0.33	5
Q4	4.76	0.60	5
Q5	4.92	0.28	5
Contenido	4.72	0.61	5
Diseño	4.85	0.43	5
General	4.80	0.51	5

En general, podemos concluir que estas descripciones lingüísticas generadas son muy adecuadas tanto en contenido como en forma, con medias de 4.72 y 4.85 respectivamente, para describir la distribución del índice de calidad del aire en las 20 estaciones.

A partir de los resultados tan positivos obtenidos en la validación por expertos, este sistema está actualmente en producción y será próximamente desplegado en la página web oficial de MeteoGalicia como un nuevo servicio público para los usuarios que consulten la información sobre la calidad del aire.

V. CONCLUSIONES

En este trabajo presentamos un modelo de generación de lenguaje natural empleando cuantificadores borrosos para la generación automática de descripciones lingüísticas a partir de datos numéricos obtenidos de un caso real de aplicación en el campo de la información medioambiental, proporcionando explicaciones textuales sobre la distribución de los valores del índice de calidad del aire (ICA), que es un indicador muy conocido proporcionado por todas las agencias meteorológicas del mundo. Basado en este modelo, describimos ICA2Text, un sistema D2T para la generación automática de descripciones en lenguaje natural sobre los datos del ICA proporcionados por la Agencia de Meteorología de Galicia.

Los resultados de la validación manual por parte de los expertos meteorólogos muestran que las descripciones en lenguaje natural generadas son muy adecuadas, puesto que en media calificaron las descripciones en lenguaje natural generadas por el sistema ICA2Text con un 4.72 sobre 5 en una escala Likert en cuanto a calidad del contenido y un 4.85 sobre 5 en cuanto a calidad lingüística (diseño).

Actualmente, ICA2Text se ha integrado en la web de producción de MeteoGalicia, de modo que tras un periodo de pruebas, como trabajo futuro se plantea el despliegue como servicio público en la sección de calidad del aire de la página web de MeteoGalicia.

Agradecimientos

Este trabajo fue financiado por el Ministerio de Ciencia, Innovación y Universidades (proyecto TIN2017-84796-C2-1-R) y la Consellería de Educación, Universidad e Formación Profesional de la Xunta de Galicia (proyectos ED431C2018/29 y ED431G2019/04). Todos los proyectos fueron cofinanciados por el programa FEDER.

Referencias

[1] E. Reiter, "Has a Consensus NL Generation Architecture Appeared, and is it Psycholinguistically Plausible?" in *Proceedings of the Seventh International Workshop on Natural Language Generation, INLG 1994, Kennebunkport, Maine, USA, June 21-24, 1994*, 1994. [Online]. Available: https://www.aclweb.org/anthology/W94-0319/

[2] ——, "An architecture for data-to-text systems," in *Proceedings of the Eleventh European Workshop on Natural Language Generation*. Association for Computational Linguistics, 2007, pp. 97–104. [Online]. Available: https://doi.org/10.3115%2F1610163.1610180

[3] J. Kacprzyk and R. R. Yager, "Linguistic summaries of data using fuzzy logic," *International Journal of General System*, vol. 30, no. 2, pp. 133–154, 2001.

[4] L. A. Zadeh, "A prototype-centered approach to adding deduction capability to search engines-the concept of protoform," in *Intelligent Systems, 2002. Proceedings. 2002 First International IEEE Symposium*, vol. 1. IEEE, 2002, pp. 2–3. [Online]. Available: https://doi.org/10.1109%2Fis.2002.1044219

[5] R. R. Yager, K. M. Ford, and A. J. Cañas, "An Approach to the Linguistic Summarization of Data," in *Uncertainty in Knowledge Bases, 3rd International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, IPMU '90, Paris, France, July 2-6, 1990, Proceedings*, ser. Lecture Notes in Computer Science, vol. 521. Springer, 1990, pp. 456–468. [Online]. Available: https://doi.org/10.1007/BFb0028132

[6] L. A. Zadeh, "A New Direction in AI - Toward a Computational Theory of Perceptions," in *Computational Intelligence, Theory and Applications, International Conference, 7th Fuzzy Days, Dortmund, Germany, October 1-3, 2001, Proceedings*, ser. Lecture Notes in Computer Science, vol. 2206. Springer, 2001, p. 628. [Online]. Available: https://doi.org/10.1007/3-540-45493-4_63

[7] A. Wilbik, J. M. Keller, and G. L. Alexander, "Linguistic summarization of sensor data for eldercare," in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, Anchorage, Alaska, USA, October 9-12, 2011*. IEEE, 2011, pp. 2595–2599. [Online]. Available: https://doi.org/10.1109/ICSMC.2011.6084067

[8] J. Kacprzyk and A. Wilbik, "A comprehensive comparison of time series described by linguistic summaries and its application to the comparison of performance of a mutual fund and its benchmark," in *FUZZ-IEEE 2010, IEEE International Conference on Fuzzy Systems, Barcelona, Spain, 18-23 July, 2010, Proceedings*. IEEE, 2010, pp. 1–8. [Online]. Available: https://doi.org/10.1109/FUZZY.2010.5584881

[9] P. Conde-Clemente, J. M. Alonso, and G. Triviño, "Toward automatic generation of linguistic advice for saving energy at home," *Soft Comput.*, vol. 22, no. 2, pp. 345–359, 2018. [Online]. Available: https://doi.org/10.1007/s00500-016-2430-5

[10] A. Ramos-Soto, A. Bugarin, S. Barro, and J. Taboada, "Linguistic descriptions for automatic generation of textual short-term weather forecasts on real prediction data," *IEEE Transactions on Fuzzy Systems*, vol. 23, no. 1, pp. 44–57, 2015. [Online]. Available: https://doi.org/10.1109%2Ftfuzz.2014.2328011

[11] A. Europea de Medio Ambiente, "Página web de la Agencia Europea de Medio Ambiente," 2021, [Accedido Mayo 2021]. [Online]. Available: www.eea.europa.eu/themes/air/air-quality-index

[12] MeteoGalicia, "Página web de la agencia de meteorología gallega meteoGalicia," 2021, [Accedido Mayo 2021]. [Online]. Available: www.meteogalicia.gal

[13] S. Busemann and H. Horacek, "Generating air quality reports from environmental data," in *Proceedings of the DFKI Workshop on Natural Language Generation*, 1997, pp. 15–21.

[14] L. Wanner, B. Bohnet, N. Bouayad-Agha, F. Lareau, and D. Nicklaß, "MARQUIS: Generation of User-Tailored Multilingual Air Quality Bulletins," *Appl. Artif. Intell.*, vol. 24, no. 10, pp. 914–952, 2010. [Online]. Available: https://doi.org/10.1080/08839514.2010.529258

[15] A. Ramos-Soto, A. Bugarín, S. Barro, N. Gallego, C. Rodríguez, I. Fraga, and A. Saunders, "Automatic Generation of Air Quality Index Textual Forecasts Using a Data-To-Text Approach," in *Advances in Artificial Intelligence - 16th Conference of the Spanish Association for Artificial Intelligence, CAEPIA 2015, Albacete, Spain, November 9-12, 2015, Proceedings*, ser. Lecture Notes in Computer Science, vol. 9422. Springer, 2015, pp. 164–174. [Online]. Available: https://doi.org/10.1007/978-3-319-24598-0_15

[16] MeteoGalicia, "Página web de la calidad del aire de la agencia de meteorología gallega meteoGalicia," 2021, [Accedido Mayo 2021]. [Online]. Available: www.meteogalicia.gal/Caire

[17] H. P. Grice, "Logic and conversation," in *Speech acts*. Brill, 1975, pp. 41–58.

[18] A. Ramos-Soto, J. J. Gallardo, and A. Bugarín, "Adapting SimpleNLG to Spanish," in *Proceedings of the 10th International Conference on Natural Language Generation, INLG 2017, Santiago de Compostela, Spain, September 4-7, 2017*. Association for Computational Linguistics, 2017, pp. 144–148. [Online]. Available: https://doi.org/10.18653/v1/w17-3521

[19] A. Cascallar-Fuentes, A. Ramos-Soto, and A. Bugarín, "Adapting SimpleNLG to Galician language," in *Proceedings of the 11th International Conference on Natural Language Generation, Tilburg University, The Netherlands, November 5-8, 2018*. Association for Computational Linguistics, 2018, pp. 67–72. [Online]. Available: https://doi.org/10.18653/v1/w18-6507

[20] A. Gatt and E. Reiter, "SimpleNLG: A Realisation Engine for Practical Applications," in *ENLG 2009 - Proceedings of the 12th European Workshop on Natural Language Generation, March 30-31, 2009, Athens, Greece*. The Association for Computer Linguistics, 2009, pp. 90–93. [Online]. Available: https://www.aclweb.org/anthology/W09-0613/

[21] C. Barros, "Proposal of a Hybrid Approach for Natural Language Generation and its Application to Human Language Technologies," 2019.

[22] D. M. Howcroft, A. Belz, M.-A. Clinciu, D. Gkatzia, S. A. Hasan, S. Mahamood, S. Mille, E. van Miltenburg, S. Santhanam, and V. Rieser, "Twenty Years of Confusion in Human Evaluation: NLG Needs Evaluation Sheets and Standardised Definitions," in *Proceedings of the 13th International Conference on Natural Language Generation*. Dublin, Ireland: Association for Computational Linguistics, Dec. 2020, pp. 169–182. [Online]. Available: https://www.aclweb.org/anthology/2020.inlg-1.23

[23] C. van der Lee, A. Gatt, E. van Miltenburg, S. Wubben, and E. Krahmer, "Best practices for the human evaluation of automatically generated text," in *Proceedings of the 12th International Conference on Natural Language Generation, INLG 2019, Tokyo, Japan, October 29 - November 1, 2019*. Association for Computational Linguistics, 2019, pp. 355–368.

Evaluación empírica de modelos de cuantificación borrosa aplicados a un agente conversacional

Mariña Canabal-Juanatey, Jose M. Alonso-Moral, Alejandro Catala, Alberto Bugarín-Diz
Centro Singular de Investigación en Tecnoloxías Intelixentes (CiTIUS),
Universidade de Santiago de Compostela, *15782, Santiago de Compostela, SPAIN*
marina.canabal@rai.usc.es, {josemaria.alonso.moral, alejandro.catala, alberto.bugarin.diz}@usc.es

***Resumen*—En este trabajo presentamos un estudio empírico orientado a poner en evidencia posibles inconsistencias que pueden producirse cuando utilizamos expresiones en lenguaje natural con cuantificadores imprecisos, y éstas son evaluadas con modelos de cuantificación borrosa que incumplen dos propiedades de interés que se han definido en la bibliografía para dichos modelos (antonimia y efecto acumulativo). La experimentación considera el modelo de cuantificación de Zadeh en el contexto de un agente conversacional que realiza recomendaciones. Los resultados muestran que hay diferencias significativas en la percepción de la consistencia y la utilidad de las conversaciones mantenidas entre una persona usuaria y el agente conversacional, relacionadas con el modelo de cuantificación utilizado.**

***Index Terms*—Cuantificadores borrosos, Generación de lenguaje natural, Evaluación humana, Agentes conversacionales**

I. Introducción

La presencia de la cuantificación en el uso del lenguaje natural humano es continua y juega un papel fundamental tanto en términos de significado como de expresividad. A menudo empleamos expresiones imprecisas como *"algunos de los alumnos sacaron buenas notas"*, *"en la mayor parte de los ayuntamientos gallegos la incidencia de coronavirus es baja"* o *"habrá pocos días con cielos nublados esta semana"*, que nos permiten manejar la vaguedad del lenguaje y resumir la información de forma natural. Debido a esta constante aparición de expresiones cuantificadas en el lenguaje natural, su modelado se ha convertido en un problema de gran interés en múltiples campos relacionados con la Inteligencia Artificial.

El objetivo principal de los cuantificadores es permitirnos caracterizar y describir propiedades cuantitativas sobre un conjunto de elementos (un colectivo, denominado *referencial*), en lugar de tener que hacerlo sobre cada elemento individual. La forma en que los datos son agregados y resumidos en una sentencia cuantificada depende directamente del modelo de cuantificación que se utilice para procesar la información. En caso de no seleccionarse un modelo adecuado, podría presentarse un resultado que no fuera representativo de los datos o, directamente, incorrecto.

Para caracterizar el comportamiento de un modelo de cuantificación, en el sentido de que los resultados obtenidos supongan una interpretación coherente y apropiada de los datos descritos, varios autores han propuesto una serie de propiedades intuitivas y plausibles que, en principio, cualquier modelo debería satisfacer [1]–[3]. Sin embargo, la mayor parte de los modelos de cuantificación más conocidos y utilizados (por ejemplo, Zadeh [4], Yager [5], etc.) incumplen algunas de estas propiedades, por lo que presentan un comportamiento alejado del adecuado, y no deseado en algunos casos.

A pesar de que los modelos de cuantificación borrosa son bien conocidos desde hace tiempo (principio de los años 80 del siglo pasado), y que en diversos trabajos posteriores se ha llevado a cabo un análisis de las propiedades que cumplen a nivel teórico [1], [2], existen muy pocos estudios empíricos que, desde una perspectiva pragmática, midan el alcance o impacto del incumplimiento de dichas propiedades a nivel práctico en ámbitos concretos [6]. Este aspecto pragmático es de especial relevancia y actualidad, especialmente a partir del uso de cuantificadores en las descripciones lingüísticas de datos [7]–[9] y en los sistemas de generación de lenguaje natural [10], puesto que la pragmática es esencial en el lenguaje humano y el incumplimiento de algunas propiedades relevantes por parte de los cuantificadores puede dar lugar a inconsistencias en la operativa interna de las aplicaciones en estos u otros ámbitos que conviene hacer aflorar. La principal aportación de este trabajo se centra precisamente en aportar evidencias que ilustren el alcance del incumplimiento de estas propiedades y, de alguna manera, hagan transparente hacia los usuarios o desarrolladores su existencia.

Así, en este trabajo presentamos un estudio empírico que evidencia las posibles inconsistencias que pueden producirse en la práctica debido al incumplimiento de ciertas propiedades por parte del modelo de cuantificación utilizado. En concreto, nos centramos en el modelo de cuantificación escalar de Zadeh [4], para el cual estudiamos el alcance del incumplimiento de la propiedad de antonimia y el problema del efecto acumulativo, en un escenario de aplicación concreto. Para ello, hemos creado el asistente virtual *Quanversa*[1] [11], que desarrolla conversaciones breves sobre la predicción meteorológica en lenguaje natural a partir de datos [10], [12] que incluyen expresiones cuantificadas borrosas. Con fragmentos de estas conversaciones hemos realizado un estudio basado en cuestionarios, que nos ha permitido evaluar el impacto que tiene el efecto acumulativo y el incumplimiento de la propiedad de antonimia por parte del modelo de cuantificación de Zadeh en cuanto a la coherencia del diálogo mantenido entre el asistente y una persona usuaria, cuando se utilizan en él expresiones cuantificadas imprecisas.

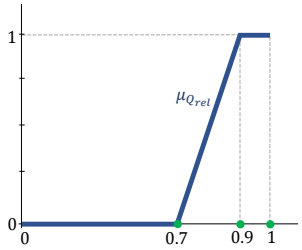
[1] https://demos.citius.usc.es/Quanversa/

El resto del trabajo está organizado en cuatro secciones. La sección II introduce brevemente los modelos de cuantificación borrosa considerados. La sección III presenta el nuevo modelo de cuantificación propuesto. La sección IV describe los experimentos realizados. Finalmente, la sección V resume las principales conclusiones.

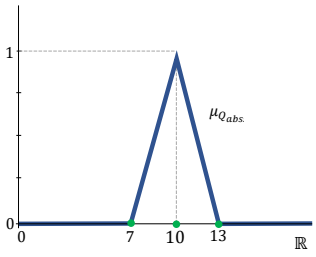
II. Modelos de Cuantificación Borrosa

En el lenguaje natural, es común encontrar términos vagos o imprecisos como "*alto*", "*joven*" o "*pequeño*", que expresan alguna propiedad sobre uno o más objetos y que pueden ser modelados mediante conjuntos borrosos. Además, la introducción de este tipo de términos dentro de expresiones cuantificadas es habitual (por ejemplo, "*muchos trabajadores son jóvenes*", "*la mayor parte de los jugadores son altos*", etc.), dónde no solo la propiedad descrita presenta incertidumbre, sino que también lo hace el cuantificador lingüístico utilizado.

Así, en el modelado de la cuantificación lingüística, el manejo de la vaguedad presente en el lenguaje es fundamental. Fue Zadeh el primero en modelar el problema de la cuantificación en el lenguaje natural utilizando la teoría de conjuntos borrosos [4]. Para ello, distinguió dos tipos de cuantificadores lingüísticos (ver ejemplos en la figura 1): *cuantificadores borrosos absolutos*, los cuales denotan una cantidad imprecisa absoluta (*"aproximadamente cinco"*, *"un número grande"*, *"alrededor de diez"*, etc.) y cuantificadores borrosos relativos (o proporcionales), que referencian cantidades relativas (*"la mitad"*, *"casi todos"*, *"una pequeña parte"*, etc.). Además, propuso identificar ambos tipos de cuantificadores con un conjunto borroso que represente la semántica del cuantificador.



(a) Cuantificador relativo Q_{rel}: *casi todos*.



(b) Cuantificador absoluto Q_{abs}: *alrededor de diez*.

Figura 1: Ejemplos de cuantificadores.

Zadeh se centró en el estudio de dos tipos de expresiones: (1) las expresiones de tipo I, que siguen el esquema (o protoforma) "Q de E son $\tilde{A}$"; y (2) las expresiones de tipo II, que responden a "Q de $(\tilde{D}E)$ son $\tilde{A}$", dónde E es el conjunto referencial, $\tilde{A}$ y $\tilde{D}$ dos conjuntos borrosos que representan alguna propiedad de E, y Q un cuantificador borroso absoluto o relativo. Un ejemplo de expresión de tipo I es "*la mayoría de estudiantes de matemáticas son altos*", mientras que un ejemplo de expresión de tipo II podría ser "*la mayoría de los buenos estudiantes de matemáticas son altos*", identificando en ambos casos E con el conjunto de estudiantes de matemáticas, Q con el cuantificador borroso relativo "*la mayoría*", $\tilde{A}$ y $\tilde{D}$ con los conjuntos borrosos que representan las propiedades de "*ser alto*" y "*ser buen estudiante*", respectivamente.

En lo que sigue, si $\tilde{A}$ es un conjunto borroso que representa una propiedad borrosa acerca de los elementos de un referencial $E = \{e_1, \ldots, e_n\}$, escribiremos $\mu_{\tilde{A}}(E) = \{a_1, \ldots, a_n\}$, con $a_i = \mu_{\tilde{A}}(e_i)$ para $i \in \{1, \ldots, n\}$, representando los grados de cumplimiento de la propiedad $\tilde{A}$ sobre los elementos de E.

Siguiendo la aproximación propuesta por Zadeh, un *modelo de cuantificación* es un método que permite combinar el conjunto borroso que representa la semántica del cuantificador Q con los conjuntos borrosos que modelan las propiedades $\tilde{A}$ y $\tilde{D}$, de forma que sea posible obtener una medida de la veracidad o grado de cumplimiento de la expresión cuantificada. El modelo de cuantificación escalar propuesto por Zadeh en [4] es uno de los métodos más conocidos y utilizados en la práctica para la evaluación de expresiones cuantificadas. En el caso de expresiones de tipo I, se define, para cuantificadores borrosos absolutos, mediante la expresión:

$$\mathcal{Z}_{Q_{abs}}(\tilde{A}) = \mu_{Q_{abs}}\left(\sum_{i=1}^{n} a_i\right), \tag{1}$$

y, para cuantificadores borrosos relativos, como:

$$\mathcal{Z}_{Q_{rel}}(\tilde{A}) = \mu_{Q_{rel}}\left(\sum_{i=1}^{n} a_i/n\right). \tag{2}$$

En el caso de expresiones de tipo II se define, para cuantificadores borrosos absolutos, como:

$$\mathcal{Z}_{Q_{abs}}(\tilde{A}/\tilde{D}) = \mu_{Q_{abs}}\left(\sum_{i=1}^{n} \text{mín}\{a_i, d_i\}\right), \tag{3}$$

y, para cuantificadores borrosos relativos, como:

$$\mathcal{Z}_{Q_{rel}}(\tilde{A}/\tilde{D}) = \mu_{Q_{rel}}\left(\frac{\sum_{i=1}^{n} \text{mín}\{a_i, d_i\}}{\sum_{i=1}^{n} d_i}\right). \tag{4}$$

A partir de este primer método introducido por Zadeh, surgieron nuevas propuestas para la evaluación de expresiones cuantificadas [5], [13], [14], de forma que cada uno utiliza un esquema diferente para obtener el grado de verdad de una expresión cuantificada. Con el objetivo de poder comparar y evaluar el comportamiento de cada uno de estos modelos, en relación a la obtención de resultados coherentes y apropiados, varios autores han propuesto una serie de propiedades que, en principio, deberían satisfacer todos aquellos modelos que tengan un comportamiento consistente, intuitivo o plausible [1].

Entre ellas, se encuentran, por ejemplo, la continuidad, la monotonía, las relacionadas con la negación o la antonimia. Sin embargo, muchos de los modelos más utilizados en la práctica no verifican algunas de estas propiedades fundamentales. En concreto, el comportamiento teórico no adecuado del modelo de Zadeh debido al incumplimiento de ciertas propiedades matemáticas ha sido estudiado en profundidad en la bibliografía [1], [2]. En este trabajo, nos centraremos únicamente en dos de sus principales problemas: el incumplimiento de la propiedad de antonimia y el efecto acumulativo.

Debido a su continuo uso en el lenguaje natural, es fundamental que los modelos se comporten correctamente respecto a las negaciones. En este sentido, la propiedad de antonimia establece que todo modelo de cuantificación Γ debe verificar:

$$\Gamma_Q(\tilde{A}) = \Gamma_{Q_{ant}}(\tilde{A}_c) \quad \text{y} \tag{5}$$

$$\Gamma_Q(\tilde{A}/\tilde{D}) = \Gamma_{Q_{ant}}(\tilde{A}_c/\tilde{D}), \tag{6}$$

siendo $\tilde{A}_c$ el complementario de $\tilde{A}$, definido por:

$$\mu_{\tilde{A}_c}(e) = 1 - \mu_{\tilde{A}}(e) \text{ para todo } e \in E$$

y Q_{ant} el antónimo de Q, que se define, si Q es un cuantificador borroso absoluto, como:

$$\mu_{Q_{ant}}(x) = \mu_Q(n - x) \text{ para todo } x \in [0, n].$$

y, si Q es un cuantificador borroso relativo, como:

$$\mu_{Q_{ant}}(x) = \mu_Q(1 - x) \text{ para todo } x \in [0, 1].$$

La violación de la propiedad de antonimia puede tener consecuencias importantes en la evaluación de expresiones cuantificadas. Por ejemplo, dos expresiones como *"todos los días del fin de semana habrá temperaturas altas"* y *"no habrá días este fin de semana en los que las temperaturas no sean altas"*, que semánticamente son equivalentes, podrían obtener grados de verdad diferentes al ser evaluadas sobre un mismo referencial. Este comportamiento poco intuitivo no debería ocurrir en ningún caso, puesto que ambas expresiones en lenguaje natural realmente significan lo mismo.

El modelo de Zadeh no verifica la propiedad anterior, como se demuestra con el siguiente contraejemplo; sean E el conjunto de los próximos 4 días, $\tilde{A}$ y $\tilde{D}$ los conjuntos borrosos que representan las propiedades "*tener temperatura alta*" y "*pertenecer al fin de semana*", de forma que $\mu_{\tilde{A}}(E) = \{0.5, 1, 1, 0.5\}$ y $\mu_{\tilde{D}}(E) = \{0.5, 1, 1, 0.5\}$. Definiendo el cuantificador $\forall :=$"*todos*" como:

$$\mu_\forall(x) = \begin{cases} 1 \text{ si } x = 1, \\ 0 \text{ si } x \neq 1, \end{cases}$$

se tiene que:

$$\mathcal{Z}_\forall(\tilde{A}/\tilde{D}) = \mu_\forall\left(\frac{0.5+1+1+0.5}{0.5+1+1+0.5}\right) = \mu_\forall(1) = 1,$$

mientras que:

$$\mathcal{Z}_{\forall_{ant}}(\tilde{A}_c/\tilde{D}) = \mu_{\forall_{ant}}\left(\frac{0.5+0+0+0.5}{0.5+1+1+0.5}\right) =$$

$$= \mu_{\forall_{ant}}(1/3) = \mu_\forall(1 - 1/3) = 0.$$

Por otro lado, el modelo de Zadeh puede provocar un efecto acumulativo poco intuitivo, ya que es posible que varios elementos con un grado de pertenencia bajo alcancen el mismo valor acumulado que un único elemento con un grado de pertenencia alto. Por ejemplo, supongamos que E y E' son los conjuntos de temperaturas para los próximos 5 días en dos ayuntamientos diferentes y $\tilde{A}$ es el conjunto borroso que representa la propiedad de "*ser una temperatura baja*" de forma que $\mu_{\tilde{A}}(E) = \{0.2, 0.2, 0.2, 0.2, 0.2\}$ y $\mu_{\tilde{A}}(E') = \{1, 0, 0, 0, 0\}$. Definiendo el cuantificador absoluto "*al menos uno*" como:

$$\mu_{\text{"al menos uno"}}(x) = \begin{cases} 0 \text{ si } x < 1, \\ 1 \text{ si } x \geq 1, \end{cases}$$

y utilizando la fórmula (1), la evaluación de la expresión *"al menos una temperatura será alta"* mediante el modelo de Zadeh devolverá el mismo resultado para ambos conjuntos referenciales E y E', pero la situación en cada ayuntamiento es diferente.

A pesar de los problemas evidenciados a nivel teórico, no se conoce el impacto que puede tener el uso de un modelo de cuantificación que presente estas deficiencias durante la interacción entre un agente conversacional y sus usuarios.

III. Modelo de Zadeh modificado

Hemos propuesto una modificación del modelo de Zadeh para corregir los potenciales efectos indeseados asociados a la antonimia y el efecto acumulativo [11], y de este modo disponer de dos modelos de cuantificación muy similares que permitiesen evaluar qué impacto tiene el incumplimiento de las propiedades mencionadas en un contexto de aplicación real. Se realizaron las modificaciones imprescindibles para abordar directamente las dos propiedades citadas.

Por un lado, a la hora de realizar una agregación, el modelo modificado solo considera los elementos que tengan un grado de pertenencia suficientemente alto. Esto significa que, en las definiciones (1)−(4), son excluidos aquellos términos de los sumatorios en los que los valores de a_i o d_i no alcancen el 0.5, evitando la acumulación de pequeños grados de pertenencia en la agregación. Se ha escogido el valor 0.5 por ser el umbral que determina la pertenencia al conjunto cuando la variable es desborrosificada. Así, los datos incluidos en la agregación serán aquellos considerados en todo momento como pertenecientes al conjunto.

Por otro lado, el incumplimiento de la propiedad de antonimia se debe a la incorrecta evaluación de una expresión cuantificada cuando ésta se formula en términos de $\tilde{A}_c$. Dado que semánticamente las expresiones "Q de $(\tilde{D}E)$ son $\tilde{A}_c$" y "Q_{ant} de $(\tilde{D}E)$ son $\tilde{A}$" son equivalentes, el modelo modificado realiza un procesamiento previo a la evaluación para transformar la primera en la segunda, cuando es necesario. Así, las expresiones de (5) y (6) se cumplen trivialmente y el modelo verifica la propiedad de antonimia.

En el estudio empírico que describimos en la siguiente sección utilizaremos este modelo de Zadeh modificado junto con el original para la evaluación de proposiciones cuantificadas borrosas sobre un mismo referencial. Se plantean escenarios de

conversación entre el agente conversacional *Quanversa* [11] y una persona usuaria en los cuales las propiedades de antonimia y efecto acumulativo juegan un papel relevante. Ello nos permitirá comparar, desde el punto de vista pragmático, el comportamiento del modelo modificado frente al original, y valorar experimentalmente la importancia en un escenario concreto del incumplimiento de las propiedades mencionadas.

IV. Estudio Empírico

De un modo más formal, el estudio plantea un experimento que permita verificar la siguiente hipótesis de investigación: *"La interacción con un agente conversacional resulta más consistente y de mayor utilidad para los usuarios cuando los datos son procesados mediante un modelo de cuantificación que supera los problemas de la antonimia y el efecto acumulativo propios del modelo de cuantificación escalar propuesto originalmente por Zadeh [4]"*.

Con el fin de validar esta hipótesis, se construyó un cuestionario que permitió que los participantes en el estudio pudieran valorar la consistencia y la utilidad de una serie de conversaciones reales entre el agente conversacional *Quanversa* y una usuaria ficticia, María, mostradas mediante una captura de pantalla (véase una muestra de un escenario en la figura 2). El cuestionario incluye 6 escenarios distintos, que nos permiten considerar las diferentes situaciones o contextos de estudio en base al modelo considerado, tamaño de la muestra, y la propiedad a analizar (ver la figura 3). Se somete a los participantes a dichos escenarios (o estímulos), sin permitir interacción en tiempo real con el asistente, para evitar efectos de otros factores no controlados que no son objeto de estudio, y así dar respuesta a nuestra hipótesis de partida. El cuestionario se distribuyó a través de listas de distribución y redes sociales, y se mantuvo abierto desde el 6 hasta el 26 de abril de 2021. Durante este tiempo, fue completado por 132 personas, de forma totalmente voluntaria y anónima.

La figura 4 muestra la relación entre el modelo de cuantificación utilizado por *Quanversa* durante la conversación y la consistencia en sus respuestas apreciada por los participantes. Las conversaciones asociadas al nuevo modelo se consideraron más consistentes que en el caso del modelo original.

La figura 5 representa la relación entre el modelo de cuantificación y la utilidad de la conversación para la resolución de la tarea propuesta en el escenario, según los participantes en el estudio. De nuevo, se observan diferencias entre ambos modelos, aunque menos acentuadas que para el caso de la consistencia.

Para contrastar la hipótesis de investigación planteada (con un nivel de significación de 0.01), basta con comprobar la dependencia o independencia entre las variables respuesta y el modelo de cuantificación. Las hipótesis nula y alternativa para los contrastes de independencia entre el modelo de cuantificación y la consistencia/utilidad en cada contexto de estudio (véase Ci, con $i \in \{1, 2, 3\}$, en la figura 3) son:

H_0**:** *La variable respuesta consistencia/utilidad es independiente del factor modelo de cuantificación en* C_i.

H_a**:** *El factor modelo de cuantificación influye en la variable respuesta consistencia/utilidad en* C_i.

Teniendo en cuenta lo anterior, se ha seleccionado el test de McNemar [15], un test de independencia no paramétrico utilizado como la alternativa a los test χ^2 de Pearson cuando los datos son pareados. El test de McNemar estudia si la probabilidad de evento positivo para una variable (en nuestro caso, conversación consistente o conversación útil) es igual en los dos niveles de otra variable (en nuestro caso, para los dos modelos de cuantificación). Su estadístico sigue una distribución χ^2 con 1 grado de libertad.

El test de McNemar aplicado a los tres contextos de estudio revela que existen diferencias significativas, que soportan rechazar H_0 en favor de que sí existe relación entre el modelo de cuantificación y la consistencia de las respuestas dadas por *Quanversa* (C1: $\chi^2 = 73.11$, p-valor< 0.01; C2: $\chi^2 = 81.01$, p-valor< 0.01; C3: $\chi^2 = 100.08$, p-valor< 0.01).

Análogamente, comprobamos la independencia de la variable utilidad respecto al modelo de cuantificación. De nuevo, para los tres contextos estudiados, el test de McNemar arroja diferencias significativas, rechazando H_0 en favor de que sí existe relación entre el modelo de cuantificación y la utilidad de la conversación mostrada entre *Quanversa* y la usuaria *María* (C1: $\chi^2 = 24.32$, p-valor< 0.01; C2: $\chi^2 = 37.21$, p-valor< 0.01; C3: $\chi^2 = 39.2$, p-valor< 0.01).

V. Conclusiones

La aplicación del test no paramétrico de McNemar revela que las diferencias encontradas son significativas tanto en la percepción de la consistencia como en la percepción de la utilidad respecto al modelo de cuantificación considerado. Junto con la información recogida en las figuras 4 y 5, se puede afirmar que un porcentaje significativo de participantes encontró las interacciones con el agente conversacional más consistentes y de mayor utilidad cuando este procesó los datos mediante el modelo de cuantificación de Zadeh modificado (que cumple la propiedad de antonimia y supera el problema del efecto acumulativo). Por lo tanto, el experimento realizado valida la hipótesis de investigación planteada.

Como trabajo futuro nos planteamos ampliar el estudio a contextos de tipo conversacional en otros ámbitos y a otros modelos de cuantificación, de modo que se pueda extender la base experimental del presente estudio y generalizar las conclusiones que hemos presentado en este trabajo. Asimismo, estudiaremos con mayor profundidad dos aspectos pragmáticos que hemos observado en el análisis de resultados, y cuyo alcance merece atención y requiere investigación adicional. Por un lado, el hecho de que, con cierta frecuencia, los usuarios fueron capaces de tomar una decisión en relación a la tarea propuesta incluso en escenarios donde habían detectado inconsistencias en la conversación. Por otro, el hecho de que un buen número de participantes tardaron más en responder a los escenarios que involucraban al modelo de Zadeh, lo cual puede ser debido a la inconsistencia en la conversación que producía dicho modelo.

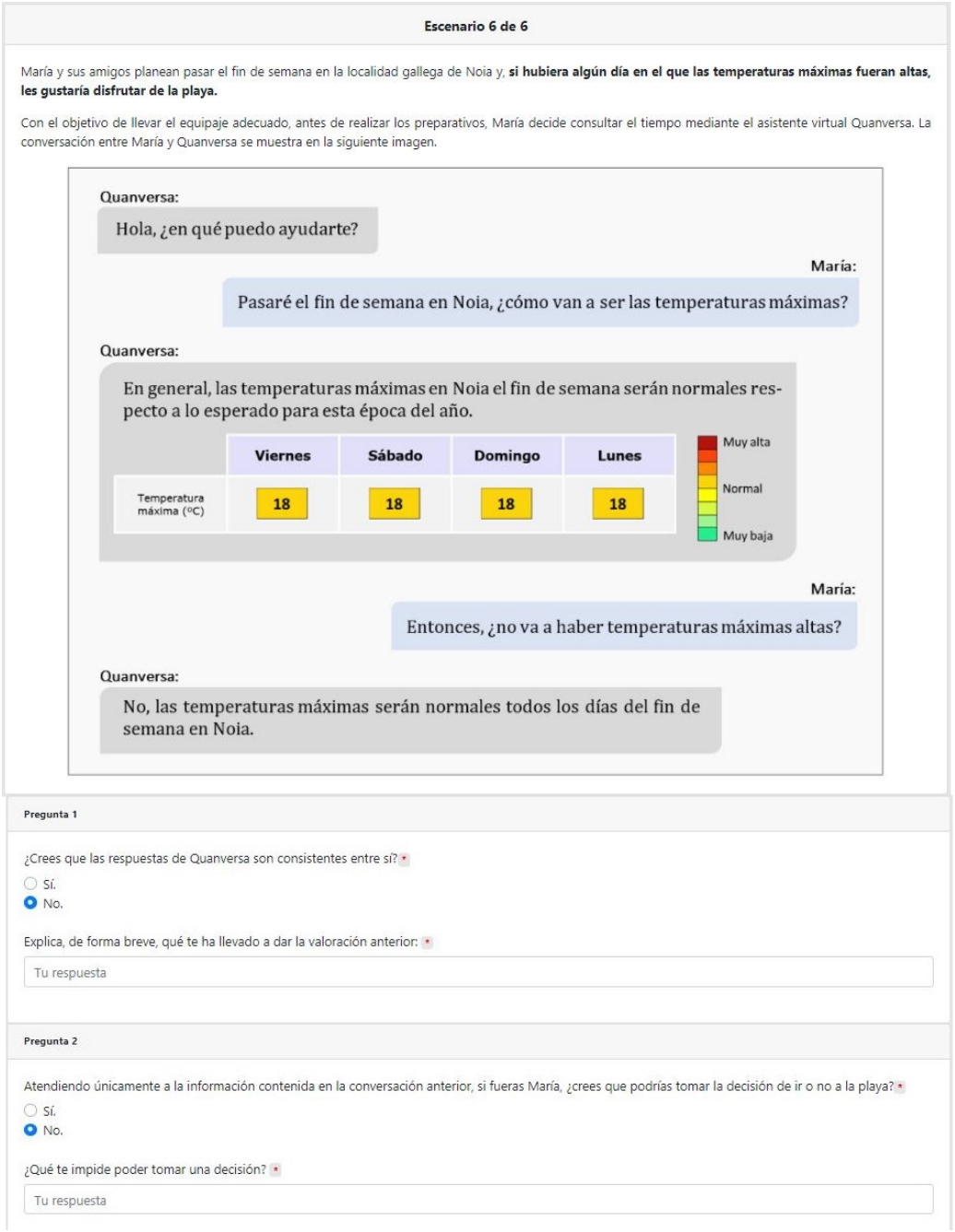


Figura 2: Ejemplo de escenario a evaluar en el cuestionario.

Agradecimientos

Jose M. Alonso-Moral es investigador *Ramón y Cajal* (RYC-2016-19802). Alejandro Catalá es investigador *Juan de la Cierva* (IJC2018-037522-I). Este trabajo está financiado por el Ministerio de Ciencia, Innovación y Universidades (proyectos TIN2017-84796-C2-1-R y RTI2018-099646-B-I00) y por la Consellería de Educación, Universidade e Formación Profesional de la Xunta de Galicia (proyectos ED431F2018/02, ED431C2018/29, y ED431G2019/04). Todos los proyectos han sido confinanciados por el Fondo Europeo de Desarrollo Regional (FEDER).

Referencias

[1] S. Barro, A. Bugarín, P. Cariñena, and F. Díaz-Hermida, "A framework for fuzzy quantification model analysis," *IEEE Transactions on Fuzzy Systems*, vol. 11, no. 1, pp. 89–99, 2003.

[2] M. Delgado, M. D. Ruiz, D. Sánchez, and M. A. Vila, "Fuzzy quantification: a state of the art," *Fuzzy Sets Syst.*, vol. 242, pp. 1–30, 2014.

[3] I. Glöckner, *Fuzzy Quantifiers: A computational Theory*. Springer, 2006.

	Modelo de Zadeh		Modelo modificado	
	Efecto acumulativo	Antonimia	Efecto acumulativo	Antonimia
Muestra pequeña	Escenario 1 (C1)	Escenario 3 (C3)	Escenario 4 (C1)	Escenario 6 (C3)
Muestra grande	Escenario 2 (C2)		Escenario 5 (C2)	

Figura 3: Escenarios seleccionados para el estudio.

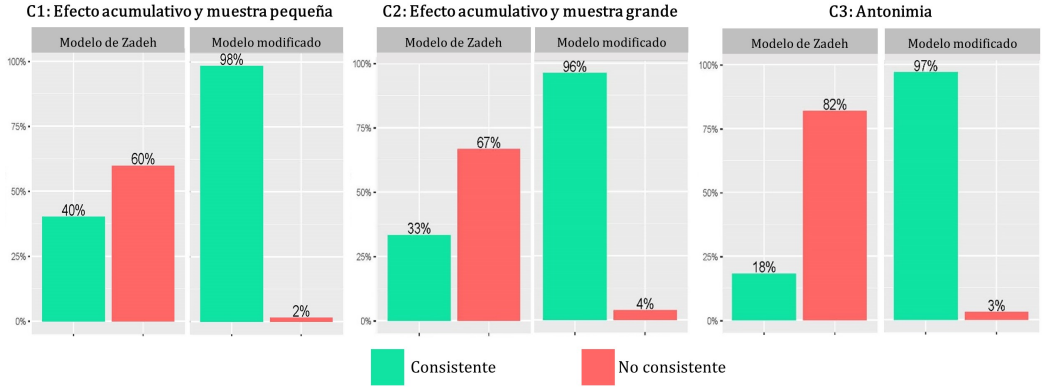


Figura 4: Relación entre la consistencia de la conversación y el modelo de cuantificación.

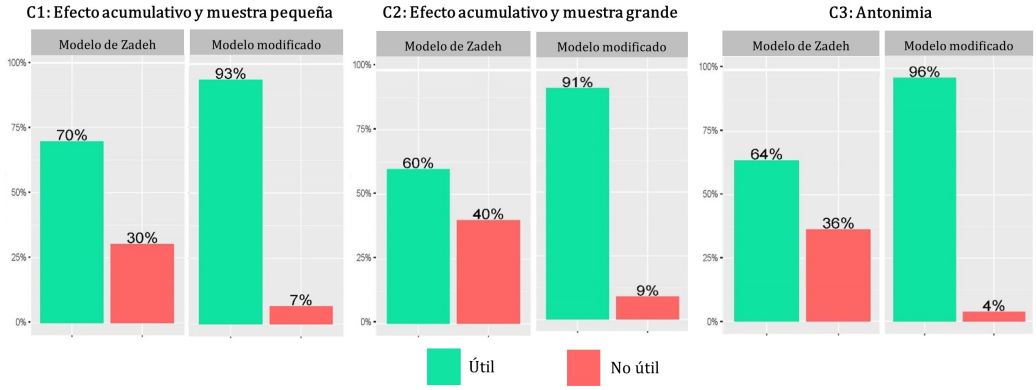


Figura 5: Relación entre la utilidad de la conversación y el modelo de cuantificación utilizado.

[4] L. A. Zadeh, "A computational approach to fuzzy quantifiers in natural languages," *Computers & Mathematics with Applications*, vol. 9, no. 1, pp. 149–184, 1983.

[5] R. R. Yager, "On ordered weighted averaging aggregation operators in multicriteria decisionmaking," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 19, no. 1, pp. 183–191, 1988.

[6] A. Cascallar-Fuentes, A. Ramos-Soto, and A. Bugarín-Diz, "An experimental study on the use of fuzzy quantification models for linguistic descriptions of data," in *Proc. 24th European Conference on Artificial Intelligence*. IOS Press, 2020, pp. 267–274.

[7] P. Conde-Clemente, J. M. Alonso, and G. Triviño, "Toward automatic generation of linguistic advice for saving energy at home," *Soft Comput.*, vol. 22, no. 2, pp. 345–359, 2018.

[8] J. Kacprzyk and A. Wilbik, "A comprehensive comparison of time series described by linguistic summaries and its application to the comparison of performance of a mutual fund and its benchmark," in *Proc. IEEE International Conference on Fuzzy Systems*, 2010, pp. 1–8.

[9] N. Marín and D. Sánchez, "On generating linguistic descriptions of time series," *Fuzzy Sets and Systems*, vol. 285, pp. 6–30, 2016, special Issue on Linguistic Description of Time Series.

[10] A. Ramos-Soto, A. Bugarín, S. Barro, and J. Taboada, "Linguistic descriptions for automatic generation of textual short-term weather forecasts on real prediction data," *IEEE Transactions on Fuzzy Systems*, vol. 1, no. 23, pp. 44–57, 2015.

[11] M. Canabal-Juanatey, *Quanversa: Modelos de cuantificación borrosa aplicados a un agente conversacional*. Universidad de Santiago de Compostela, Trabajo Fin de Grado, Ingeniería Informática, 2021.

[12] A. Ramos-Soto, J. M. Alonso, E. Reiter, K. van Deemter, and A. Gatt, "Fuzzy-based language grounding of geographical references: From writers to readers," *International Journal of Computational Intelligence Systems*, vol. 12, no. 2, pp. 970–983, 2019.

[13] P. Bosc and L. Lietard, "Monotonic quantified statements and fuzzy integrals," in *NAFIPS/IFIS/NASA Conference*, 1994, pp. 8–12.

[14] M. Delgado, D. Sánchez, and M. A. Vila, "Fuzzy cardinality based evaluation of quantified sentences," *International journal of Approximate Reasoning*, vol. 23, pp. 23–6, 2000.

[15] Q. McNemar, "Note on the sampling error of the difference between correlated proportions or percentages," *Psychometrika*, vol. 12, no. 2, pp. 153–157, 1947.

Hypothesis Scoring and Model Refinement Strategies for FM-based RANSAC*

*Note: The full contents of this paper have been published in the volume Lecture Notes in Artificial Intelligence 12882 (LNAI 12882)

Alberto Ortiz, Esaú Ortiz, Juan José Miñana, Óscar Valero
Dep. Mathematics and Computer Science
University of the Balearic Islands, and IDISBA (Institut d'Investigacio Sanitaria de les Illes Balears)
Palma de Mallorca, Spain
{alberto.ortiz,esau.ortiz,jj.minana,o.valero}@uib.es

***Abstract*—Robust model estimation is a recurring problem in application areas such as robotics and computer vision. Taking inspiration from a notion of distance that arises in a natural way in fuzzy logic, this paper modifies the well-known robust estimator RANSAC making use of a Fuzzy Metric (FM) within the estimator main loop to encode the compatibility of each sample to the current model/hypothesis. Further, once a number of hypotheses have been explored, this FM-based RANSAC makes use of the same fuzzy metric to refine the winning model. The incorporation of this fuzzy metric permits us to express the distance between two points as a kind of degree of nearness measured with respect to a parameter, which is very appropriate in the presence of the vagueness or imprecision inherent to noisy data. By way of illustration of the performance of the approach, we report on the estimation accuracy achieved by FM-based RANSAC and other RANSAC variants for a benchmark comprising a large number of noisy datasets with varying proportion of outliers and different levels of noise. As it will be shown, FM-based RANSAC outperforms the classical counterparts considered.**

***Index Terms*—Model estimation, RANSAC, Fuzzy metric, 2D straight line estimation**

DOI: https://doi.org/10.1007/978-3-030-85713-4_10

Análisis de distintos tipos de tendencias en sucesiones de contextos L-fuzzy

Cristina Alcalde
Dep. Matemática Aplicada
Universidad del País Vasco - UPV/EHU
San Sebastián
c.alcalde@ehu.eus

Ana Burusco
Dep. Estadística, Informática y Matemáticas
Universidad Pública de Navarra
Instituto de Smart Cities
Pamplona
burusco@unavarra.es

***Resumen*—En este trabajo estudiamos el crecimiento, decrecimiento o ausencia de tendencias en las relaciones entre objetos y atributos en una sucesión de contextos L-fuzzy. Estas tendencias nos permitirán ordenar los objetos, atributos y conceptos atendiendo a su crecimiento o decrecimiento a lo largo del tiempo. Finalmente, ilustraremos nuestros resultados con una aplicación práctica.**

***Palabras Clave*—Conceptos L-Fuzzy, sucesiones de contextos L-Fuzzy, análisis de tendencias.**

I. Introducción

El Análisis de Conceptos L-Fuzzy [1], [2] es una herramienta matemática para el análisis y la representación del conocimiento conceptual. Esta teoría utiliza los conceptos L-fuzzy para extraer información de un contexto L-fuzzy. Recordemos que un contexto L-fuzzy es una tupla (L, X, Y, R), donde L es un retículo completo, X e Y son conjuntos de objetos y atributos, y $R \in L^{X\times Y}$ es una relación L-fuzzy entre los objetos y los atributos. Podemos entender el Análisis de Conceptos L-fuzzy como una extensión del Análisis de Conceptos Formales de Wille [3], [4] que permite trabajar con relaciones entre los objetos y atributos que toman valores en un retículo L, en lugar de valores binarios.

Para trabajar con estos contextos L-fuzzy, hemos definido los operadores derivación 1 y 2 por medio de las expresiones:

Para todo $A \in L^X$, para todo $B \in L^Y$

$$A_1(y) = \inf_{x\in X}\{I(A(x), R(x, y))\}, \forall y \in Y$$

$$B_2(x) = \inf_{y\in Y}\{I(B(y), R(x, y))\}, \forall x \in X$$

con I un operador de implicación fuzzy definido en el retículo $(L, \leq)$.

La información almacenada en el contexto se visualiza por medio de los conceptos L-fuzzy, que representan a un grupo de objetos que comparten, ellos y sólo ellos, un grupo de atributos. Estos conceptos son pares $(M, M_1) \in L^X \times L^Y$, donde el conjunto $M \in fix(\varphi)$ es un punto fijo del operador constructor φ, que se define a partir de los operadores derivación 1 y 2 como $\varphi(M) = (M_1)_2 = M_{12}$.

Trabajo parcialmente subvencionado por el Gobierno Vasco (Proyecto IT1256-19), y por el Grupo de Investigación "Inteligencia Artificial y Razonamiento Aproximado" de la Universidad Pública de Navarra.

Además, fijado un conjunto de partida $A \in L^X$ (o $B \in L^Y$), podemos obtener el concepto L-fuzzy asociados aplicando sucesivamente los operadores derivación hasta encontrar un punto fijo.

Para el caso de utilizar implicaciones residuadas, tal y como haremos en este trabajo, obtendremos un punto fijo en la segunda aplicación de los operadores derivación, con lo que el cálculo del concepto L-fuzzy asociado a un conjunto de partida se simplifica enormemente. Así, dado un conjunto de objetos $A \in L^X$ (o un conjunto de atributos $B \in L^Y$), el concepto L-fuzzy asociado será (A_{12}, A_1) (o (B_2, B_{21})).

En este trabajo trataremos de estudiar en qué medida las relaciones entre los objetos y los atributos mejoran o empeoran con el paso del tiempo. Representaremos estas situaciones mediante una secuencia de contextos L-fuzzy en la que profundizaremos en el estudio de las tendencias tanto crecientes como decrecientes. También estudiaremos los casos en los que no haya tendencia.

Existen trabajos en la literatura que analizan la evolución temporal en un contexto formal, por ejemplo, [5], [6]. En particular, en [6], Wolff introduce un Sistema de Tiempo Conceptual para definir el Análisis de Conceptos Temporales. En este Sistema de Tiempo Conceptual, el estado y la fase se definen como retículos de conceptos que representan el significado de los estados con respecto a la descripción elegida del tiempo. Además, otros autores definen tendencias de evolución en [5], utilizando temporal matching en el caso del Análisis de Conceptos Formales.

II. Tendencias temporales

En [7] presentamos un primer estudio de sucesiones de contextos L-fuzzy considerando el retículo $L = [0, 1]$. Comenzaremos este apartado recordando cómo definíamos estas sucesiones:

Definición 1: Una sucesión de contextos L-fuzzy es una sucesión de tuplas (L, X, Y, R_i), $i \in \{1, 2, \ldots, n\}$, siendo L un retículo completo, X e Y dos conjuntos de objetos y atributos respectivamente y para cada valor $i \in \{1, \ldots, n\}$ $R_i \in L^{X\times Y}$ representa la relación entre los conjuntos X e Y en el instante i, la cual toma valores en el retículo L.

Posteriormente, con el objeto de estudiar la evolución de la relación entre los objetos (o atributos) respecto de uno o

varios atributos (u objetos), en [8], [9] realizamos un análisis de tendencias temporales para identificar la evolución con el tiempo de la sucesión de contextos L-fuzzy $(L, X, Y, R_i), i \in \{1, \ldots, n\}$.

Una buena herramienta para analizar la evolución en el tiempo de un objeto o un atributo es el estudio de sus conceptos L-fuzzy asociados en los diferentes contextos L-fuzzy de la sucesión. Ésta es la idea en la que se basan las siguientes definiciones:

Definición 2: Consideremos el objeto $x_0 \in X$. Sean $(A_{i\{x_0\}}, B_{i\{x_0\}})$ los conceptos L-fuzzy asociados a $\{x_0\}$ en la sucesión de contextos L-fuzzy (L, X, Y, R_i) con $i \leq n$. Denotamos por $ITrend(x_0)$ al conjunto de atributos cuya relación con el objeto x_0 se va fortaleciendo en la sucesión de contextos, y lo definimos como:

$$\begin{aligned} ITrend(x_0) =& \{y \in Y \mid B_{1\{x_0\}}(y) \neq B_{n\{x_0\}}(y) \\ & \wedge B_{i\{x_0\}}(y) \leq B_{i+1\{x_0\}}(y), \forall i < n\} \end{aligned}$$

Análogamente, dado el atributo $y_0 \in Y$, y los conceptos L-fuzzy $(A_{i\{y_0\}}, B_{i\{y_0\}})$ asociados a $\{y_0\}$ en la sucesión de contextos L-fuzzy, definiremos el conjunto $ITrend(y_0)$ que representa el conjunto de objetos que están cada vez más relacionados con el atributo y_0 de la siguiente manera:

$$\begin{aligned} ITrend(y_0) =& \{x \in X \mid A_{1\{y_0\}}(x) \neq A_{n\{y_0\}}(x) \\ & \wedge A_{i\{y_0\}}(x) \leq A_{i+1\{y_0\}}(x), \forall i < n\} \end{aligned}$$

En el caso de tendencias decrecientes, podemos definir los conjuntos de objetos o atributos cuyos grados de relación con un determinado atributo u objeto sea cada vez menor.

Definición 3: Fijado el objeto $x_0 \in X$ definimos el conjunto

$$\begin{aligned} DTrend(x_0) =& \{y \in Y \mid B_{1\{x_0\}}(y) \neq B_{n\{x_0\}}(y) \\ & \wedge B_{i\{x_0\}}(y) \geq B_{i+1\{x_0\}}(y), \forall i < n\} \end{aligned}$$

formado por los atributos cuyo grado de relación con dicho objeto va disminuyendo con el paso del tiempo.

Del mismo modo, para el atributo $y_0 \in Y$ obtenemos el conjunto de objetos cada vez menos relacionados con él:

$$\begin{aligned} DTrend(y_0) =& \{x \in X \mid A_{1\{y_0\}}(x) \neq A_{n\{y_0\}}(x) \\ & \wedge A_{i\{y_0\}}(x) \geq A_{i+1\{y_0\}}(x), \forall i < n\} \end{aligned}$$

Finalmente, tendremos que tener en cuenta que podría haber situaciones en las cuales no se observe una tendencia creciente ni decreciente. Para representar estos casos definimos los siguientes conjuntos:

Definición 4: El conjunto de objetos que no muestran ninguna tendencia en relación con el objeto $x_0 \in X$ es el siguiente:

$$NTrend(x_0) = \{y \in Y / y \notin ITrend(x_0) \cup DTrend(x_0)\}$$

Para el atributo $y_0 \in Y$, el conjunto de objetos sin ningún tipo de tendencia se define como:

$$NTrend(y_0) = \{x \in X / x \notin ITrend(y_0) \cup DTrend(y_0)\}$$

Es sencillo demostrar que, dados un objeto y un atributo cualesquiera del contexto L-fuzzy, se cumple la siguiente propiedad.

Proposición 1: Para todo par de elementos $(x, y) \in X \times Y$ se verifica que:

1. $x \in ITrend(y) \Longleftrightarrow y \in ITrend(x)$
2. $x \in DTrend(y) \Longleftrightarrow y \in DTrend(x)$

Estas definiciones de conjuntos de objetos (o de atributos) que presentan algún tipo de tendencia en su variación en el tiempo respecto de un atributo (o de un objeto) dado, permitirán establecer pares de objetos y atributos que pueden ser usados para un análisis más completo de la evolución de la sucesión de contextos L-fuzzy $(L, X, Y, R_i), i \in \{1, \ldots, n\}$.

A partir de esta idea, se construyen las *matrices de tendencias* en el contexto L-fuzzy que indican la relación entre un objeto y un atributo cuando se observa algún tipo de tendencia relacionada con ellos.

Definición 5: La matriz de tendencia creciente $ITM \subseteq X \times Y$ se define como (ver [10]):

$$\begin{aligned} & ITM(x, y) = \\ & = \begin{cases} 1 & \text{si } y \in ITrend(x) \text{ (o equival. } x \in ITrend(y)) \\ 0 & \text{en otro caso} \end{cases} \end{aligned}$$

De forma análoga podemos definir las matrices de tendencia decreciente y de ausencia de tendencia de la siguiente manera:

$$DTM(x, y) = \begin{cases} 1 & \text{si } x \in DTrend(y) \\ 0 & \text{en otro caso} \end{cases}$$

$$NTM(x, y) = \begin{cases} 1 & \text{si } y \in NTrend(x) \\ 0 & \text{en otro caso} \end{cases}$$

Estas matrices de tendencias nos permiten ahora definir los contextos formales (X, Y, ITM), (X, Y, DTM) y (X, Y, NTM) y, a partir del análisis de sus respectivos conceptos formales, tener una visión general de las tendencias entre los objetos X y los atributos Y.

Definición 6: Sea el contexto formal (X, Y, ITM) con X conjunto de objetos, Y conjunto de atributos y $ITM \subseteq X \times Y$. Llamaremos a los conceptos de (X, Y, ITM) *conceptos formales* $ITrend$. Estos conceptos representarán las tendencias crecientes.

De la misma manera, podemos obtener los *conceptos formales* $DTrend$ para las tendencias decrecientes y *conceptos formales* $NTrend$ para los casos de ausencia de tendencias.

El cálculo de las matrices de tendencia nos va a permitir analizar aquellos objetos y atributos que mejoran o empeoran su relación con el tiempo. Sin embargo, no dispondremos de herramientas para cuantificar lo que supone esa mejora o empeoramiento.

En [10] y con el fin de medir el grado de evolución positiva de las relaciones entre los objetos y los atributos, se definía su nivel de tendencia como un valor en el intervalo $[0, 1]$. Esa manera de definir el nivel de tendencia tiene el inconveniente de que recoge el incremento que ha habido en los valores

de la relación entre un objeto y un atributo, pero no la relevancia absoluta del valor. Por este motivo, presentamos aquí una nueva definición para los niveles de tendencias que nos proporcionará una información más precisa.

Dado que los análisis de las tendencias crecientes y de las decrecientes son en muchos aspectos simétricos, por simplificar, escribiremos $\Lambda Trend$ para referirnos a cualquiera de las tendencias $ITrend$ o $DTrend$:

Definición 7: Para cada $x_0 \in X, y \in Y$, el nivel $\Lambda TrendL$ del objeto x_0 para el atributo y se define como:

$$\Lambda TrendL(x_0)_y = \\ = \begin{cases} [\min\limits_{1\leq i\leq n} B_{i\{x_0\}}(y), \max\limits_{1\leq i\leq n} B_{i\{x_0\}}(y)] & \text{si } y \in \Lambda Trend(x_0) \\ 0 & \text{en otro caso} \end{cases}$$

Análogamente, podemos definir para cada $y_0 \in Y, x \in X$ el nivel $\Lambda TrendL$ del atributo y_0 para el objeto x:

$$\Lambda TrendL(y_0)_x = \\ = \begin{cases} [\min\limits_{1\leq i\leq n} A_{i\{y_0\}}(x), \max\limits_{1\leq i\leq n} A_{i\{y_0\}}(x)] & \text{si } x \in \Lambda Trend(y_0) \\ 0 & \text{en otro caso} \end{cases}$$

Proposición 2: Para todo par de elementos $(x, y) \in X \times Y$ se verifica que:

$$\Lambda TrendL(x)_y = \Lambda TrendL(y)_x$$

A partir de la definición de los niveles de tendencias podemos establecer nuevas relaciones entre los objetos y los atributos de la siguiente manera:

Definición 8: Denominamos *relación de* Λ*-tendencia* a la siguiente relación intervalovalorada definida entre los conjuntos de objetos atributos.

Para cada $(x, y) \in X \times Y$,

$$\Lambda TrendLM(x, y) = \Lambda TrendL(x)_y = \Lambda TrendL(y)_x$$

III. Rankings de objetos, atributos y conceptos atendiendo a las tendencias

A partir de las definiciones de $ITrendLM$ y $DTrendLM$ para el caso de crecimiento y decrecimiento, vamos a intentar definir relaciones entre los objetos y los atributos de acuerdo con su evolución en el tiempo.

Denotaremos por $\underline{ITrendLM}(x, y)$ y $\overline{ITrendLM}(x, y)$, respectivamente, al extremo inferior y superior de intervalo $ITrendLM(x, y)$. Análogamente para $DTrendLM(x, y)$.

Definición 9: Para cada $x \in X, y \in Y$, el porcentaje de crecimiento $ITrendLPer$ y de decrecimiento $DTrendLPer$ del objeto x para el atributo y se definen como:

$$ITrendLPer(x, y) = \\ = \begin{cases} \frac{\overline{ITrendLM}(x,y) - \underline{ITrendLM}(x,y)}{1 - \underline{ITrendLM}(x,y)} & \text{si } x \in ITrend(y) \\ 0 & \text{en otro caso} \end{cases}$$

Del mismo modo,

$$DTrendLPer(x, y) = \\ = \begin{cases} \frac{\overline{DTrendLM}(x,y) - \underline{DTrendLM}(x,y)}{\overline{DTrendLM}(x,y)} & \text{si } x \in DTrend(y) \\ 0 & \text{en otro caso} \end{cases}$$

Agregando los porcentajes de crecimiento y, análogamente, de decrecimiento de cada objeto o atributo podremos analizar sus niveles porcentuales de tendencia.

Definición 10: Para cada $x_0 \in X, y_0 \in Y$ definimos su *nivel porcentual de tendencia creciente (o decreciente)* de la siguiente manera:

$$\Lambda TLP(x_0) = \underset{y\in Y}{Agr}(\Lambda TrendLPer(x_0, y)) \\ \Lambda TLP(y_0) = \underset{x\in X}{Agr}(\Lambda TrendLPer(x, y_0))$$

donde Agr es un operador de agregación.

Esta definición nos permite establecer relaciones de orden en los conjuntos de objetos y de atributos en función de sus niveles de tendencia:

Definición 11: Dados $z_i, z_j \in X$ o $z_i, z_j \in Y$, definimos las relaciones $<_{\Lambda TL}$ como $z_i <_{\Lambda TL} z_j$ si $\Lambda TLP(z_i) < \Lambda TLP(z_j)$.

De este modo, obtendremos rankings de crecimiento si trabajamos con $ITLP$. Aquellos objetos o atributos con mayores valores de $ITLP$ serán los que hayan evolucionado de forma más positiva a lo largo del tiempo dentro de su posibilidad de mejora. También los habrá de decrecimiento, si lo hacemos con $DTLP$.

Para obtener una información más completa sobre las tendencias que se pueden observar en la secuencia de contextos L-fuzzy, el siguiente paso consistirá en analizar la evolución conjunta de objetos y atributos.

Así, utilizando las matrices de tendencias ΛTM, estableceremos relaciones en los conceptos $Trend$ definidos a partir de los contextos formales $(X, Y, \Lambda TM)$ que nos permitan establecer qué conceptos son los que presentan unas tendencias más acentuadas.

Asignaremos, en primer lugar, un nivel de tendencia porcentual a cada uno de los conceptos formales de la siguiente manera:

Definición 12: Para cada (A, B) concepto formal del contexto formal $(X, Y, \Lambda TM)$ se define el nivel porcentual de Λ-tendencia del concepto como:

$$\Lambda TLPC(A, B) = \\ = \begin{cases} \underset{(x,y)\in A\times B}{Agr}(\Lambda TrendLPer(x, y)) & \text{si } A, B \neq \emptyset \\ 0 & \text{en otro caso} \end{cases}$$

donde Agr es un operador de agregación.

Esto nos permite obtener relaciones entre los conceptos formales de los contextos definidos a partir de las matrices de tendencias y establecer qué conceptos son los que presentan tendencias crecientes o decrecientes más acentuadas. Para ello utilizaremos la siguiente relación de orden:

Definición 13: Dados $(A,B),(C,D) \in X \times Y$ definimos la relación $<_{\Lambda TLC}$ de modo que $(A,B) <_{\Lambda TLC} (C,D)$ si $\Lambda TLPC(A,B) < \Lambda TLPC(C,D)$.

IV. Aplicación práctica

Para ilustrar la aplicación de los resultados vamos a considerar una sucesión de contextos L-fuzzy $(L,X,Y,R_i), i \in \{1,\ldots,5\}$ que recoge la estancia media de los clientes en distintos tipos de alojamientos turísticos en algunas regiones de España a lo largo de un periodo de cinco años.

El objetivo del estudio será analizar si ha habido algún tipo de tendencia creciente o decreciente en dichas estancias, teniendo en cuenta tanto los tipos de alojamientos como las regiones consideradas en el estudio.

En estos contextos L-fuzzy hemos considerado el conjunto de objetos formado por los distintos tipos de alojamientos X ={x_1=Hotel, x_2=Camping, x_3=Apartamento Turístico, x_4=Casa Rural }. Los atributos considerados son las distintas regiones en las que se ha realizado el estudio Y = {y_1=Andalucía, y_2=Cataluña, y_3=Navarra, y_4=País Vasco} y los valores de las relaciones se corresponden con las estancias medias registradas desde 2016 hasta 2019. Los valores se han normalizado para trabajar con el retículo formado por la cadena $L = \{0, 0.01, \ldots, 0.99, 1\}$ (ver Tabla I).

Analizando los conjuntos derivados obtenidos a partir de los distintos objetos o atributos, se obtienen las matrices de tendencia creciente ITM y de tendencia decreciente DTM que se muestran, respectivamente, en la Tabla II y Tabla III. Estas matrices nos permiten observar para qué objetos hemos tenido algún tipo de tendencia y en cuáles de los atributos se ha mantenido dicha tendencia.

Podemos observar, por ejemplo, que mientras la estancia media en hoteles (x_1) tuvo una tendencia creciente en Andalucía (y_1), en Navarra y País Vasco (y_3 e y_4) la tendencia fue decreciente.

Tras calcular las distintas matrices de tendencias, con el fin de conocer en qué regiones y en qué tipos de alojamientos se ha dado una tendencia más acentuada, vamos a establecer rankings entre los objetos y los atributos. Calculamos para ello los niveles porcentuales de tendencia tanto creciente como decreciente. Los valores obtenidos se muestran en la Tabla IV.

Estos niveles de tendencia nos permiten ordenar los objetos y atributos atendiendo a su mayor crecimiento o decrecimiento. La Figura 1 muestra la ordenación obtenida desde el objeto (o atributo) con una mayor tendencia creciente.

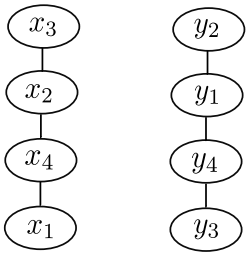


Figura 1. Rankings de objetos y atributos con tendencia creciente.

Podemos observar que el tipo de alojamiento en el que más ha crecido la estancia media ha sido en los apartamentos turísticos (x_3), mientras que la región que ha tenido una mayor subida en sus estancias vacacionales ha sido Cataluña (y_2).

Análogamente, observando el ranking obtenido a partir de los niveles de tendencia decreciente (ver Figura 2), vemos que el tipo de establecimiento que ha acusado una mayor bajada ha sido el camping (x_2). En cuanto a las regiones, la que ha sufrido un mayor descenso ha sido Navarra (y_3).

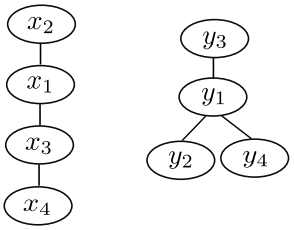


Figura 2. Rankings de objetos y atributos con tendencia decreciente.

Con el fin de tener una visión conjunta de alojamientos y regiones, a partir de las matrices de tendencia mostradas en la Tabla II, definimos los contextos formales (X,Y,ITM) e (X,Y,DTM) y obtenemos los niveles de tendencia asociados a sus distintos conceptos formales.

Para los conceptos formales del contexto (X,Y,ITM) se obtienen los niveles porcentuales de tendencia creciente:

$$ITLPC(X,\emptyset) = 0$$
$$ITLPC(\{x_1,x_3,x_4\},\{y_1\}) = 0{,}25$$
$$ITLPC(\{x_2,x_3\},\{y_2\}) = 0{,}93$$
$$ITLPC(\{x_4\},\{y_1,y_3\}) = 0{,}42$$
$$ITLPC(\{x_3\},\{y_1,y_2,y_4\}) = 0{,}54$$
$$ITLPC(\emptyset,Y) = 0$$

A partir de estos niveles de tendencia obtenemos la prelación en el conjunto de conceptos, ordenados de mayor a menor crecimiento, representada en la Figura 3. Tal y como podemos

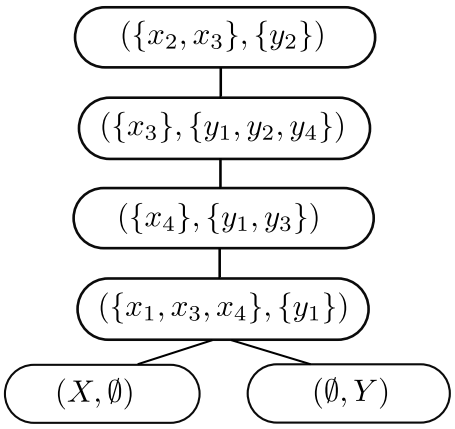


Figura 3. Ranking de conceptos con tendencia creciente.

Tabla I
RELACIONES DE LA SUCESIÓN DE CONTEXTOS L-FUZZY.

2015

R_1	y_1	y_2	y_3	y_4
x_1	0.38	0.40	0.31	0.35
x_2	0.79	0.52	0.80	0.41
x_3	0.47	0.80	0.56	0.36
x_4	0.51	0.47	0.40	0.41

2016

R_2	y_1	y_2	y_3	y_4
x_1	0.39	0.41	0.29	0.33
x_2	0.63	0.64	0.58	0.37
x_3	0.49	0.80	0.54	0.42
x_4	0.52	0.40	0.46	0.39

2017

R_3	y_1	y_2	y_3	y_4
x_1	0.39	0.46	0.29	0.30
x_2	0.58	0.65	0.46	0.36
x_3	0.56	0.83	0.54	0.55
x_4	0.56	0.39	0.49	0.38

2018

R_4	y_1	y_2	y_3	y_4
x_1	0.43	0.44	0.22	0.30
x_2	0.52	0.83	0.45	0.42
x_3	0.56	0.87	0.47	0.59
x_4	0.60	0.37	0.52	0.42

2019

R_5	y_1	y_2	y_3	y_4
x_1	0.47	0.42	0.18	0.28
x_2	0.50	1	0.45	0.39
x_3	0.58	0.97	0.35	0.72
x_4	0.71	0.37	0.66	0.43

Tabla II
MATRIZ DE TENDENCIA CRECIENTE.

ITM	y_1	y_2	y_3	y_4
x_1	1	0	0	0
x_2	0	1	0	0
x_3	1	1	0	1
x_4	1	0	1	0

Tabla III
MATRIZ DE TENDENCIA DECRECIENTE.

DTM	y_1	y_2	y_3	y_4
x_1	0	0	1	1
x_2	1	0	1	0
x_3	0	0	1	0
x_4	0	1	0	0

observar, las estancias que han mantenido una mayor tendencia creciente se han dado en campings y apartamentos turísticos (x_2 y x_3) de Cataluña (y_2), seguidas de las estancias en apartamentos turísticos (x_3) de Andalucía, Cataluña y País Vasco (y_1, y_2 e y_4).

De manera análoga, podemos calcular los niveles porcentuales de tendencia para los conceptos formales del contexto (X, Y, DTM) definido a partir de la matriz de tendencia decreciente. Los valores obtenidos son los siguientes:

$$DTLPC(X, \emptyset) = 0$$
$$DTLPC(\{x_4\}, \{y_2\}) = 0{,}21$$
$$DTLPC(\{x_1, x_2, x_3\}, \{y_3\}) = 0{,}41$$
$$DTLPC(\{x_2\}, \{y_1, y_3\}) = 0{,}40$$
$$DTLPC(\{x_1\}, \{y_3, y_4\}) = 0{,}31$$
$$DTLPC(\emptyset, Y) = 0$$

Tabla IV
NIVELES PORCENTUALES DE TENDENCIA.

$ITLP$

x_1	0.04	y_1	0.19
x_2	0.25	y_2	0.46
x_3	0.41	y_3	0.11
x_4	0.21	y_4	0.14

$DTLP$

x_1	0.15	y_1	0.09
x_2	0.20	y_2	0.05
x_3	0.09	y_3	0.31
x_4	0.05	y_4	0.05

La ordenación de los conceptos de mayor a menor decrecimiento es la representada en la Figura 4.

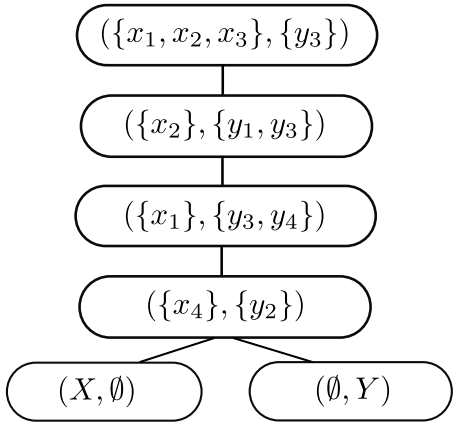


Figura 4. Ranking de conceptos con tendencia decreciente.

Observamos aquí que el mayor decrecimiento se ha dado en las estancias medias en hoteles, campings y apartamentos turísticos (x_1, x_2 y x_3) de Navarra (y_3).

V. CONCLUSIONES Y TRABAJO FUTURO

Hemos continuado con el estudio de tendencias en una sucesión de contextos L-fuzzy que representa la evolución en el tiempo de un contexto L-fuzzy analizando las tendencias

crecientes, decrecientes y la ausencia de tendencias. Se ha analizado la relevancia de dichas tendencias temporales lo que ha permitido establecer rankings de objetos, atributos y conceptos formales. Como línea futura se estudiarán clasificaciones de objetos y de atributos a partir de conjuntos de partida y según su crecimiento o decrecimiento.

REFERENCIAS

[1] A. Burusco, R. Fuentes-González: The Study of the L-fuzzy Concept Lattice, *Mathware and Soft Computing* 1 (3), pp. 209–218, 1994.

[2] A. Burusco, R. Fuentes-González: Construction of the L-fuzzy Concept Lattice, *Fuzzy Sets and Systems* 97 (1), pp. 109–114, 1998.

[3] B. Ganter, R. Wille: Formal concept analysis: Mathematical foundations, Springer, Berlin - New York, 1999.

[4] R. Wille: Restructuring lattice theory: an approach based on hierarchies of concepts, in: *Rival I. (Ed.), Ordered Sets, Reidel, Dordrecht-Boston*, pp. 445-470, 1982.

[5] R. Neouchi, A. Y. Tawfik, R. A. Frost: Towards a Temporal Extension of Formal Concept Analysis. In Proceedings of Canadian Conference on Artificial Intelligence, pp. 335–344, 2001.

[6] K. E. Wolff: Temporal Relational Semantic Systems, *ICCS*, pp. 165–180, 2010.

[7] C. Alcalde, A. Burusco, R. Fuentes-González: The study of fuzzy context sequences *Int. Journal of Computational Intelligence Systems* 6 (3), pp. 518–529, 2013.

[8] C. Alcalde, A. Burusco: L-fuzzy context sequences on complete lattices, IPMU 2014, *Lecture Notes in Computer Science* Part III, CCIS 444, pp. 31–40, 2014.

[9] C. Alcalde, A. Burusco, H. Bustince, A. Jurio, J.A. Sanz: Evolution in time of the L-fuzzy context sequences, *Information Sciences* 326, pp. 202-214, 2016.

[10] C. Alcalde, A. Burusco: Estudio sobre la evolución de la relación entre objetos y atributos en una sucesión de contextos L-fuzzy, XVIII Conferencia de la Asociación Española para la Inteligencia Artificial, CAEPIA 2018, Granada, pp. 273–278, 2018.

On the standard fuzzy metric: generalizations and application to model estimation

Juan José Miñana, Alberto Ortiz, Esaú Ortiz and Óscar Valero
Department of Mathematics and Computer Science, *University of the Balearic Islands* and IDISBA, Palma, Spain
{jj.minana, alberto.ortiz, esau.ortiz, o.valero}@uib.es

***Abstract*—Different approaches to obtain a notion of metric in the context of fuzzy setting can be found in the literature. In this paper, we deal with the concept due to George and Veeramani, which is defined by means of continuous triangular norms. Different authors have addressed the study of such a concept from a theoretical point of view. In this paper, we provide a new methodology to induce fuzzy metrics which generalize the celebrated standard fuzzy metric. The aforementioned methodology allows us to approach some questions related to the continuous triangular norms from which such fuzzy metrics are defined. Moreover, we show the applicability of the new fuzzy metrics to an engineering problem. More specifically, we address successfully robust model estimation through a variant of the well-known estimator RANSAC. By way of illustration of the performance of the approach, we report on the accuracy achieved by the new estimator and other RANSAC variants for a benchmark involving a specific model estimation problem and a large number of datasets with varying proportion of outliers and different levels of noise. The resulting estimator is shown able to outperform the classical counterparts considered.**

***Index Terms*—Fuzzy metric; continuous t-norm; Dombi t-norm; standard fuzzy metric; model estimation; RANSAC**

I. Introduction and preliminaries

In 1965, L. A. Zadeh introduced the notion of fuzzy set in [1]. Since then, such a concept has constituted the grounds of many lines of research in different fields, such as Mathematics, Computer Science, Economics. In Mathematics and, in particular, in Topology, an interesting issue consists in providing a notion of metric, in the fuzzy setting, in accordance with the essence of the classical concept. With this aim, in [2], I. Kramosil and J. Michalek introduced a notion of fuzzy metric space by adapting the concept of statistical metric due to Menger (see [3]) to the fuzzy context. Later on, in [4], A. George and P. Veeramani slightly modified the notion of Kramosil and Michalek with the aim of obtaining a more faithful adaptation to the fuzzy setting of the classical concept of metric. In both cases, the concept of fuzzy metric is defined by means of continuous t-norms (see [5] to find a deep treatment on t-norms). Following [4], a *fuzzy metric space* is a triplet $(X, M, *)$ where X is a non-empty set, $*$ is a continuous t-norm and M is a fuzzy set on $X \times X \times]0, \infty[$ satisfying, for each $x, y, z \in X$ and $t, s \in]0, \infty[$, the following:

This work is partially supported by EU-H2020 projects BUGWRIGHT2 (GA 871260) and ROBINS (GA 779776), and by projects PGC2018-095709-B-C21 (MCIU/AEI/FEDER, UE), and PROCOE/4/2017 (Govern Balear, 50% P.O. FEDER 2014-2020 Illes Balears). This publication reflects only the authors views and the European Union is not liable for any use that may be made of the information contained therein.

(GV1) $M(x, y, t) > 0$;
(GV2) $M(x, y, t) = 1$ if and only if $x = y$;
(GV3) $M(x, y, t) = M(y, x, t)$;
(GV4) $M(x, z, t + s) \geq M(x, y, t) * M(y, z, s)$;
(GV5) The assignment $M_{x,y} :]0, \infty[\to]0, 1]$ is a continuous function.

As usual, we say that $(M, *)$, or simply M if no confusion arises, is a fuzzy metric on X.

On account of the previous definition, the value of $M(x, y, t)$ can be interpreted as a degree of nearness between the point x and y of X with respect to the parameter $t \in]0, \infty[$. Then, the closer to 1 is such a value, the nearer the points x and y with respect to t are. Contrarily, values close to 0 indicate a lower degree of nearness. Thus, in this notion of fuzzy metric, 1 plays a similar role to 0 for the classical case, whereas 0 can be seen as ∞ in classical metrics. So, axiom (GV1) is justified by the fact that the degree of nearness with respect to a parameter never can be zero, just as in the classical case the distance between two points cannot become ∞.

One can easily identify (GV2), (GV3) and (GV4) as fuzzy versions of the axioms of, respectively, separation, symmetry and transitivity, which altogether define the notion of classical metric. Concretely, (GV2) means that, on the one hand, the degree of nearness between two points with respect to an arbitrary parameter only can be 1 whenever both points are the same. On the other hand, the degree of nearness between a point and itself is 1, with respect to an arbitrary parameter. Finally, (GV5) ensures that no drastic changes arise in the degree of nearness due to slight modifications of the parameter with respect to which it is being measured.

An immediate consequence of (GV4), which was pointed out by M. Grabiec for fuzzy metrics in the sense of Kramosil and Michaleck (see [6]), is that the degree of nearness between two points does not decrease when the parameter for which such a degree is relative increases, i.e. for each $x, y \in X$, we have that $M(x, y, t) \geq M(x, y, s)$ for each $t, s \in]0, \infty[$ with $t > s$.

This kind of fuzzy metric spaces has been studied by several authors from the mathematical point of view. Besides, they have been used successfully in engineering problems such as colour image filtering or perceptual colour difference (see [7]–[11]). Indeed, fuzzy metrics show some advantages with respect to the classical ones. On the one hand, the parameter t allows the fuzzy metric to be better adapted to context in which it is to be used. On the other hand, fuzzy

metrics match perfectly with the employment of other fuzzy techniques, since the value given by a fuzzy metric, as pointed out before, can be directly interpreted as a fuzzy degree of nearness. So, providing useful techniques for generating fuzzy metrics becomes an interesting issue in order to provide a wider range of measurement tools in such a way that the fuzzy metric that best fits the problem being studied can be applied to solve it.

A celebrated example of fuzzy metric is the so-called standard fuzzy metric, which is defined from a classical metric (see [4]). Indeed, let (X, d) be a metric space and define the fuzzy set M_d on $X \times X \times]0, \infty[$ as follows:

$$M_d(x, y, t) = \frac{t}{t + d(x, y)}, \text{ for each } x, y \in X, t \in]0, \infty[. \tag{1}$$

The *standard fuzzy metric* on X deduced from d is the pair $(M_d, *_M)$, where $*_M$ denotes the minimum t-norm (i.e. $a *_M b = \min\{a, b\}$ for each $a, b \in [0, 1]$).

Observe that $(X, M_d, *)$ is also a fuzzy metric space for each continuous t-norm $*$, since $*_M$ is the largest t-norm. Indeed, given a continuous t-norm $*$, the inequality $a *_M b \geq a * b$ is satisfied for each $a, b \in [0, 1]$.

From the topological point of view, the standard fuzzy metric enjoys outstanding properties. The topologies generated from the standard fuzzy metric and from the classical metric, from which it is induced, coincide. Besides, it fulfils some interesting properties which do not make sense in the classical context but they do in the fuzzy context. Among others, it should be stressed the property of being strong (see [12]). Let us recall that a fuzzy metric space $(X, M, *)$ is said to be *strong* if, in addition, M satisfies, for each $x, y, z \in X$ and $t \in]0, \infty[$, the next inequality:

$$M(x, z, t) \geq M(x, y, t) * M(y, z, t). \tag{2}$$

Observe that the preceding inequality is stronger than that given in the axiom (GV4).

It is a well-known fact that, given a metric space (X, d), then the standard fuzzy metric space $(X, M_d, *_P)$ is strong, where $*_P$ denotes the usual product t-norm, i.e. $a *_P b = a \cdot b$ for each $a, b \in [0, 1]$. Nevertheless, $(X, M_d, *_M)$ is not a strong fuzzy metric space in general, as pointed out in [12]. In view of the preceding fact, an interesting question arises: there exists a continuous t-norm $*$, different from $*_P$, with $* \geq *_P$ and such that $(X, M_d, *)$ is a strong fuzzy metric space for each metric space (X, d)?

In [13], a generalization of the fuzzy set M_d given by (1) was introduced defining, for each $x, y \in X$ and $t \in]0, \infty[$, the next fuzzy set:

$$M_d^{g,m}(x, y, t) = \frac{g(t)}{g(t) + m \cdot d(x, y)}, \tag{3}$$

where $m \in]0, \infty[$ and $g :]0, \infty[\to]0, \infty[$ is a non-decreasing continuous function. According to [13], $(X, M_d^{g,m}, *_P)$ is a strong fuzzy metric space. Nevertheless, an extra condition on g is required so that $(X, M_d^{g,m}, *_M)$ is a fuzzy metric space for any arbitrary metric space (X, d). Indeed, if the function g is not superadditive, i.e. $g(t + s) \geq g(t) + g(s)$ for each $t, s \in]0, \infty[$, then $(X, M_d^{g,m}, *_M)$ is not, in general, a fuzzy metric space. Again, similar to the case of the standard fuzzy metric, it seems natural to wonder whether there exits a continuous t-norm, different from $*_P$, with $* \geq *_P$ such that $(X, M_d^{g,m}, *)$ is a fuzzy metric space for each metric space (X, d) without requiring any extra condition on g.

Coming back to the applicability of fuzzy metrics, in most problems, we are interested in measuring some kind of difference or similarity between objects. Therefore, fuzzy metrics can be good candidates to evaluate such a measurement. Concretely, the fuzzy set M is used to provide the aforementioned difference or similarity. However, the t-norm that defines M as a fuzzy metric does not play any role in the way in which such a measure is provided and, thus, it does not contribute anything that can make the fuzzy metric better fit for the problem under consideration. Since the fuzzy set $M_d^{g,m}$ given by expression (3) depends on more elements than the standard fuzzy metric M_d, $M_d^{g,m}$ allows to get more flexibility to obtain a measurement tool that fits better to the problem under consideration than M_d. So, providing a fuzzy set that generalizes expression (3) could improve the potential applicability of fuzzy metrics, even though such a generalization does not become a fuzzy metric for the same class of t-norms for which $M_d^{g,m}$ is so.

In the light of the exposed facts, the aim of this paper is twofold. On the one hand, we focus our efforts on obtaining a fuzzy set that generalizes expression (3) and on finding a family of continuous t-norms for which this new fuzzy set becomes a fuzzy metric. Moreover, we are interested in the study of those continuous t-norms for which this new fuzzy metric fulfils the property of being strong. Such a study allows us to approach the two questions posed above. On the other hand, we address a model estimation problem as an example of engineering application to illustrate the applicability of the new fuzzy metric proposed in Section II.

II. The Generalized Standard Fuzzy Metric

In this section, we build a new fuzzy metric which generalizes, in some sense, the standard fuzzy metric and the fuzzy metric given by expression (3). To this end, we recall a well-known family of continuous t-norms introduced by J. Dombi in [14].

Given $\lambda \in]0, \infty[$ the t-norm $*_{Dom}^{\lambda}$ is defined, for each $a, b \in [0, 1]$, by the following expression:

$$a *_{Dom}^{\lambda} b = \begin{cases} 0, & \text{if } a = 0 \text{ or } b = 0 \\ \dfrac{1}{1 + \left(\left(\frac{1-a}{a} \right)^{\lambda} + \left(\left(\frac{1-b}{b} \right)^{\lambda} \right) \right)^{\frac{1}{\lambda}}} & \text{otherwise} \end{cases}. \tag{4}$$

The construction of the promised fuzzy metric can be found in the next result:

Theorem 2.1: Let (X, d) be a metric space, $m, n \in]0, \infty[$ and $g :]0, \infty[\to]0, \infty[$ be a non-decreasing continuous function. Define the fuzzy set $\tilde{M}_d^{g,m,n}$ on $X \times X \times]0, \infty[$ as:

$$\tilde{M}_d^{g,m,n}(x, y, t) = \frac{g(t)}{g(t) + m \cdot d^n(x, y)}, \tag{5}$$

where $d^n(x,y)$ denotes $(d(x,y))^n$. Then, $(X, \tilde{M}_d^{g,m,n}, *)$ is a fuzzy metric space for each continuous t-norm $*$ satisfying $* \leq *_{Dom}^{\frac{1}{n}}$.

Proof.

Let $*$ be a continuous t-norm such that $* \leq *_{Dom}^{\frac{1}{n}}$. It is not hard to check that $\tilde{M}_d^{g,m,n}$ satisfies axioms (GV1), (GV2), (GV3) and (GV5). It remains to prove that (GV4) also holds.

Let $x, y, z \in X$ and $t, s \in]0, \infty[$. We will see that

$$\tilde{M}_d^{g,m,n}(x,z,t+s) \geq \tilde{M}_d^{g,m,n}(x,y,t) * \tilde{M}_d^{g,m,n}(y,z,s).$$

Set $\alpha = \max\{g(t), g(s)\}$. Observe that

$$\tilde{M}_d^{g,m,n}(x,z,t+s) \geq \frac{\alpha}{\alpha + m \cdot d^n(x,z)},$$

$$\tilde{M}_d^{g,m,n}(x,y,t) \leq \frac{\alpha}{\alpha + m \cdot d^n(x,y)}$$

and

$$\tilde{M}_d^{g,m,n}(y,z,s) \leq \frac{\alpha}{\alpha + m \cdot d^n(y,z)}.$$

So, since $* \leq *_{Dom}^{\frac{1}{n}}$, we have that

$$\tilde{M}_d^{g,m,n}(x,y,t) * \tilde{M}_d^{g,m,n}(y,z,s) \leq$$

$$\leq \tilde{M}_d^{g,m,n}(x,y,t) *_{Dom}^{\frac{1}{n}} \tilde{M}_d^{g,m,n}(y,z,s) \leq$$

$$\leq \frac{\alpha}{\alpha + m \cdot d^n(x,y)} *_{Dom}^{\frac{1}{n}} \frac{\alpha}{\alpha + m \cdot d^n(y,z)} =$$

$$= \frac{1}{1 + \frac{m \cdot (d(x,y)+d(y,z))^n}{\alpha}} = \frac{\alpha}{\alpha + m \cdot (d(x,y)+d(y,z))^n} \leq$$

$$\leq \frac{\alpha}{\alpha + m \cdot d^n(x,z)} \leq \tilde{M}_d^{g,m,n}(x,z,t+s).$$

Therefore, for each $x, y, z \in X$ and $t, s \in]0, \infty[$, $\tilde{M}_d^{g,m,n}$ satisfies (GV4) for $*$ and we conclude that $(X, \tilde{M}_d^{g,m,n}, *)$ is a fuzzy metric space. □

It must be stressed that $(X, \tilde{M}_d^{g,m,n}, *)$ is not a fuzzy metric space, in general, when $*$ does not satisfy the condition $* \leq *_{Dom}^{\frac{1}{n}}$, as the next example shows.

Example 2.2: Let $(\mathbb{R}, d_u)$ be the metric space where d_u is the Euclidean metric on $\mathbb{R}$, i.e. $d_u(x,y) = |x-y|$. Consider the non-decreasing continuous function $g_1 :]0, \infty[\to]0, \infty[$ given by $g(t) = 1$, for each $t \in]0, \infty[$, and $m = n = 1$. Then, the fuzzy set $\tilde{M}_{d_u}^{g_1,1,1}$ is given by expression (5) as follows:

$$\tilde{M}_{d_u}^{g_1,1,1}(x,y,t) = \frac{1}{1 + d_u(x,y)}, \text{ for each } x, y \in \mathbb{R}, t \in]0, \infty[.$$

Let $*$ be a continuous t-norm such that $* \not\leq *_{Dom}^{1}$. Then, there exists $a, b \in]0, 1[$ such that $a * b > a *_{Dom}^{\frac{1}{n}} b$.

Consider $x = \frac{a-1}{a}$, $y = 0$, $z = \frac{1-b}{b}$ and $t, s \in]0, \infty[$. Then,

$$\tilde{M}_{d_u}^{g_1,1,1}(x,z,t+s) = \frac{1}{1 + \frac{1-a}{a} + \frac{1-b}{b}} = a *_{Dom}^{1} b,$$

$$\tilde{M}_{d_u}^{g_1,1,1}(x,y,t) = a \text{ and } \tilde{M}_{d_u}^{g_1,1,1}(y,z,s) = b.$$

Therefore, $\tilde{M}_{d_u}^{g_1,1,1}(x,y,t) * \tilde{M}_{d_u}^{g_1,1,1}(y,z,s) = a * b > a *_{Dom}^{1} b = \tilde{M}_{d_u}^{g_1,1,1}(x,z,t+s)$, and so $\tilde{M}_{d_u}^{g_1,1,1}$ does not satisfy (GV4).

On account of Theorem 2.1 and the preceding example, we conclude that $*_{Dom}^{\frac{1}{n}}$ is the largest (continuous) t-norm for which $\tilde{M}_d^{g,m,n}$ is a fuzzy metric on X, for each arbitrary metric space (X,d), each non-decreasing continuous function $g :]0, \infty[\to]0, \infty[$ and each $m, n \in]0, \infty[$. Such a conclusion allows us to approach the two questions posed in Section I.

On the one hand, Theorem 2.1 introduces a generalization of the fuzzy set given by expression (3). Indeed, such a fuzzy set is obtained by considering $n = 1$ in the fuzzy set defined by expression (5), i.e. $M_d^{g,m} = \tilde{M}_d^{g,m,1}$. Besides, the aforementioned theorem establishes that $M_d^{g,m}$ is a fuzzy metric on X for each t-norm $*$ with $* \leq *_{Dom}^{1}$. This fact allows us to answer in affirmative way one of the questions that we wondered in Section I, which is whether there exists a continuous t-norm $* \geq *_P$ such that $(X, M_d, *)$ is a fuzzy metric space for each metric space (X,d) without requiring any extra condition on g.

First, observe that, for each $a, b \in]0, 1]$, we have that

$$a *_{Dom}^{1} b = \frac{1}{1 + \frac{1-a}{a} + \frac{1-b}{b}} = \frac{ab}{a + b - ab}.$$

Now,

$$\frac{ab}{a+b-ab} \geq a \cdot b \Leftrightarrow 1 \geq a + b - ab = a(1-b) + b.$$

Taking into account that $a \leq 1$ we have that $1 = 1 - b + b \geq a(1-b) + b$. So, $a *_{Dom}^{1} b \geq a *_P b$ for each $a, b \in]0, 1]$. Thus $*_{Dom}^{1} \geq *_P$ for each $a, b \in [0, 1]$, since $a *_{Dom}^{1} b = 0 = a *_P b$ whenever $a = 0$ or $b = 0$.

Hence, we have found a continuous Archimedean t-norm greater that the product t-norm $*_P$ for which $M_d^{g,m}$ is a fuzzy metric on X, for each arbitrary metric space (X,d), each non-decreasing continuous function $g :]0, \infty[\to]0, \infty[$ and each $m \in]0, \infty[$. Furthermore, on account of Example 2.2 we conclude that $*_{Dom}^{1}$ is the largest t-norm for which $(X, M^{g,1}, *_{Dom}^{1})$ is a fuzzy metric space, in general.

On the other hand, $\tilde{M}_d^{g,m,n}$ becomes the standard fuzzy metric when we consider the non-decreasing continuous function $g(t) = t$ and $m = n = 1$. Under this remark and, based on the argument exposed in the proof of Theorem 2.1, we prove the next result which will be useful to answer the first question about the standard fuzzy metric set out in Section I.

Theorem 2.3: Let (X,d) be a metric space, $m, n \in]0, \infty[$ and $g :]0, \infty[\to]0, \infty[$ be a non-decreasing continuous function. Then the fuzzy metric space $(X, \tilde{M}_d^{g,m,n}, *)$, where $\tilde{M}_d^{g,m,n}$ is given by (5), is strong for each continuous t-norm satisfying $* \leq *_{Dom}^{\frac{1}{n}}$.

Proof.

Consider a continuous t-norm $*$ such that $* \leq *_{Dom}^{\frac{1}{n}}$. By Theorem 2.1 we conclude that $(X, \tilde{M}_d^{g,m,n}, *)$ is a fuzzy metric space. It remains to show that inequality (2) holds.

Let $x, y, z \in X$ and $t, s \in]0, \infty[$. Then,

$$\tilde{M}_d^{g,m,n}(x,z,t) = \frac{g(t)}{g(t) + m \cdot d^n(x,z)} \geq$$

$$\geq \frac{g(t)}{g(t) + m \cdot (d(x,y) + d(y,z))^n} =$$

$$= \tilde{M}_d^{g,m,n}(x,y,t) *_{Dom}^{\frac{1}{n}} \tilde{M}_d^{g,m,n}(y,z,t) \geq$$

$$\tilde{M}_d^{g,m,n}(x,y,t) * \tilde{M}_d^{g,m,n}(y,z,t).$$

Hence, $(X, \tilde{M}_d^{g,m,n}, *)$ is a strong fuzzy metric space. □

As a consequence of the previous theorem, we conclude that $(X, M_d, *_{Dom}^1)$ is a strong fuzzy metric space. This fact answers affirmatively the first question lay out in Section I by providing a continuous t-norm greater that the product t-norm $*_P$ for which the standard fuzzy metric is strong for any arbitrary metric space (X, d).

Even more, on account of Example 2.2 we conclude that the standard fuzzy metric is just a strong fuzzy metric in general for continuous t-norms less than $*_{Dom}^1$. Notice that the aforementioned example provides that the standard fuzzy metric $(\mathbb{R}, M_{d_u}, *)$ is not strong if $* \not\leq *_{Dom}^1$. Indeed, let $a, b \in]0, 1[$ such that $a * b > a *_{Dom}^1 b$. Then, take $x = \frac{a-1}{a}$, $y = 0$ and $z = \frac{1-b}{b}$. Therefore,

$$M_{d_u}(x,z,1) = \frac{1}{1 + \frac{1-a}{a} + \frac{1-b}{b}} = a *_{Dom}^1 b < a * b =$$

$$= M_{d_u}(x,y,1) * M_{d_u}(y,z,1).$$

III. Application Case: Robust Model Estimation

Solving model estimation problems is a fundamental component of numerous applications involving perception tasks. Nowadays, facing this kind of problem requires to cope with new challenges due to an increased use of poor, low-cost sensors, and the ever growing deployment of robotic devices which may operate in potentially unknown environments. Generally speaking, the underlying algorithms have to be robust against uncertain data that besides may be corrupted by outliers, i.e. data items which are not consistent with the original model due to an arbitrary bias affecting them. A *robust estimator* is able to correctly find the original model that supposedly the input data fits to under the aforementioned conditions [15]. The Random Sample Consensus algorithm (RANSAC) [16] is one of these robust estimation techniques, which is widely used nowadays, so much that it has become common in robotics and computer vision.

Briefly speaking, RANSAC tries to achieve a maximum consensus in the input dataset in order to deduce the inliers by generating random hypotheses on the model parameters through a *hypothesize-and-verify* approach. That is to say, instead of using every sample in the dataset to perform the estimation as in traditional regression techniques, RANSAC tests many random sets of samples and outputs the one leading to the best fitting. Since picking an extra point decreases exponentially the probability of selecting an outlier-free sample [17], RANSAC takes the Minimum Sample Set size (MSS) to determine a unique candidate model, thus increasing its chances of finding an all-inlier sample set. This model is assigned a score based on the cardinality of its consensus set. Finally, RANSAC returns the hypothesis that has achieved the highest consensus and the set of inliers, which are used next to estimate the ultimate model by regression.

Searching for an all-inlier sample, RANSAC typically runs for N iterations:

$$N = \frac{\log(1-\rho)}{\log(1-(1-\omega)^s)} \tag{6}$$

where ρ is the desired probability of success, i.e. at least one of the considered random sets is outlier-free, s is the size of the MSS for the problem at hand and ω is the ratio of outliers. See [16] for the details on Eq. (6).

Algorithm 1 outlines FM-based RANSAC, a variant of RANSAC described in [18] that avoids discriminating between inliers and outliers by means of the use of a fuzzy metric that encodes as a similarity the compatibility of each sample to the currently hypothesized model. In this work, we particularize FM-based RANSAC for the fuzzy metric $\tilde{M}_d^{g,m,n}$ introduced as Eq. (5) in Section II. $\tilde{M}_d^{g,m,n}$ is also incorporated into the final model refinement step that follows the main hypothesis selection loop. Finally, in Section IV, we report on the accuracy achieved by FM-based RANSAC for a specific model estimation problem when using $\tilde{M}_d^{g,m,n}$ for different values of m and n. The assessment involves a comparison with RANSAC and MSAC [19] for a benchmark comprising a large number of datasets with varying proportion of outliers and different levels of noise.

We detail next the features of FM-based RANSAC:

1) **Samples classification**. In the original RANSAC, for every model considered, data samples are classified into inliers and outliers by comparing the fitting error with a threshold τ_I related to data noise. As already mentioned, FM-based RANSAC does not distinguish between inliers and outliers, but makes use of a compatibility value $\phi \in [0,1]$ between each sample x_j and the current model $\mathcal{M}_{\widehat{\Theta}_k}$, given the fitting error $\epsilon(x_j; \mathcal{M}_{\widehat{\Theta}_k})$. Such compatibility value derives from the fuzzy metric $\tilde{M}_d^{g,m,n}$ once parameterized by (d, Φ) with $\Phi = (n, m, g)$. Since in the following we contemplate the use of an only, specific distance d, i.e. the Euclidean metric, and g is set to the constant function θ^n as a reference of noise scale[1], we denote the fuzzy metric as $M^{m,n}$ eliminating the allusion to d and g. From now on, the value of $M^{m,n}$ will be denoted by $\phi(\epsilon; \Phi)$.
2) **Model scoring**. The individual compatibility values $\phi(\epsilon; \Phi)$ are aggregated by simple summation to obtain the model score (step 6 in Alg. 1) and hence the *so-far-the-best-model* is given by the maximum score found up to the current iteration (steps 7 - 9 of Alg. 1).

[1] In this regard, we refer to the form $M_d^{g,m,n}(x,y) = \frac{1}{1+m\cdot(d(x,y)/\theta)^n}$.

Algorithm 1 FM-based RANSAC

Input: D - dataset comprising samples $\{x_j\}$
$\phi(\epsilon\,;\,\Phi)$ - FM compatibility value for fitting error
$k_{\max}$ - maximum number of iterations of the main loop
$t_{\max}$ - maximum number of iterations of the refinement stage
Output: $\mathcal{M}_{\widehat{\Theta}}$ - estimated model

1: $k := 0,\ \varphi_{\max} := -\infty$
2: **for** $k := 1$ **to** $k_{\max}$ **do** ▷ find best consensus model $\mathcal{M}_{\widehat{\Theta}}$
3: select randomly a minimal sample set S_k of size s
4: estimate model $\mathcal{M}_{\widehat{\Theta}_k}$ from S_k
5: calculate fitting errors $\epsilon(x_j; \mathcal{M}_{\widehat{\Theta}_k}), \forall x_j \in \mathrm{D}$
6: find model score $\varphi_k := \sum_{x_j \in \mathrm{D}} \phi(\,\epsilon(x_j; \mathcal{M}_{\widehat{\Theta}_k})\,;\,\Phi\,)$
7: **if** $\varphi_k > \varphi_{\max}$ **then**
8: $\varphi_{\max} := \varphi_k,\ \mathcal{M}^0_{\widehat{\Theta}} := \mathcal{M}_{\widehat{\Theta}_k}$
9: **end if**
10: **end for**
11: $t := 0$
12: **repeat** ▷ refine model $\mathcal{M}_{\widehat{\Theta}}$
13: calculate fitting errors $\epsilon(x_j; \mathcal{M}^t_{\widehat{\Theta}}), \forall x_j \in \mathrm{D}$
14: estimate model $\mathcal{M}^{t+1}_{\widehat{\Theta}}$ using weights $\phi(\epsilon(x_j; \mathcal{M}^t_{\widehat{\Theta}})\,;\,\Phi\,)$
15: $t := t + 1$
16: **until** convergence **or** $t \geq t_{\max}$
17: **return** $\mathcal{M}^t_{\widehat{\Theta}}$

3) **Model refinement**. Once a sufficient number of models have been considered, we re-estimate the winning model using iterative weighted least squares, where the compatibility values $\phi(\epsilon;\Phi)$, calculated for the fitting errors resulting from the current model, are used as weights for the new, refined model (steps 12 - 16 of Alg. 1). The loop iterates until changes in the estimated parameters of the model $\widehat{\Theta}$ are negligible (or after $t_{\max}$ iterations).

IV. Experimental results

A. Experimental setup

For testing purposes, we consider a hyperplane model estimation problem for 2D (straight lines), 3D (planes) and 10D, the latter as a case of higher dimensionality. To this end, we generate synthetic datasets stemming from hyperplanes in random orientations and positions: 500 for 2D/3D hyperplanes and 250 for 10D hyperplanes. Given a 2D/3D/10D random point p belonging to a hyperplane with normal vector $\vec{n}$, an inlier p_I is generated by shifting p along $\vec{n}$ using a zero-mean Gaussian distribution with standard deviation σ, i.e. $p_I = p + \mathcal{N}(0,\sigma) \cdot \vec{n}$. Outliers p_O are uniformly generated within a rectangular area containing part of the hyperplane, ensuring that they lie out of a 3σ stripe at both sides of the hyperplane. Every pair (σ, ω) gives rise to a different dataset.

Regarding hypothesis generation within the main loop, in all experiments, the size of the MSS is always set to the minimum, i.e. $s = 2$, $s = 3$ and $s = 10$ for respectively 2D, 3D and 10D. Besides, the number of iterations $k_{\max}$ is calculated according to Eq. (6), with $\rho = 99\%$. The parameters of $\phi(\epsilon;\Phi)$, $\Phi = (n, m, g)$, are set as follows: $n, m \in \{1, 2\}$, as indicated for each experiment; and g is the constant function θ^n, where $\theta = \kappa \cdot \sigma$. For RANSAC/MSAC $\tau_I = \kappa \cdot \sigma$. Different values for κ are considered for both θ and τ_I. Finally, to compare properly RANSAC, MSAC and our estimator, we make use of the same sequence of MSS's to avoid the effect of randomness.

B. Results and discussion

In the following, to measure the estimation accuracy for the hyperplane fitting problem, we make use of the average $\mu[\varepsilon]$ of the angle ε between the true and the estimated normal vector. We also report on the average number of iterations spent during model refinement $\mu[t]$.

Table I and Fig. 1 show performance results for the fuzzy metric $M^{m,n}$ and several outlier ratios ω and Gaussian noise magnitudes σ. In sight of these results, it is worth noting that: (1) the estimation accuracy for $M^{2,2}$ is above that of plain RANSAC and MSAC in all cases, while, for the other configurations of $M^{m,n}$, the accuracy is in general better than the classical counterparts though not in all cases; (2) the value of θ in $M^{m,n}$ does not seem to be critical, since the highest change in $\mu[\varepsilon]$ for θ with $\kappa \in \{1, 2, 2.5, 3, 4\}$ is less than 1°; (3) the distribution of the average error $\mu[\varepsilon]$ shows always larger errors for RANSAC/MSAC than for $M^{2,2}$ for all percentiles. As for the number of iterations of the refinement stage t: (4) in general, $\mu[t]$ is similar for every combination of $M^{m,n}$ when varying the noise parameters (σ, ω) and particularly higher for $M^{2,2}$ when κ is low; (5) lower values of κ allow the proposed estimator to perform a better refinement stage in terms of accuracy but at the expense of computational cost since more iterations are required. Regarding the fuzzy metric $M^{m,n}$, both $M^{2,n}$ and $M^{m,2}$ lead in general to higher accuracy, with a slight increase in the number of refining iterations for low κ values or higher noise (σ, ω).

V. Conclusions

This work introduces a methodology to induce fuzzy metrics that generalizes the celebrated standard fuzzy metric. Moreover, some questions related to the continuous triangular norms from which such fuzzy metrics are defined have been posed and answered. A concrete new fuzzy metric induced through the aforementioned methodology has been succesfully embedded within a revised version of RANSAC. By means of this metric, we avoid discriminating between inliers and outliers, to instead make use of a compatibility value to the current model for each sample. Experimental results show good performance, actually outperforming two classical counterparts, RANSAC and MSAC.

References

[1] L. Zadeh, "Fuzzy sets," *Information and Control*, vol. 8, no. 3, pp. 338–353, 1965.
[2] I. Kramosil and J. Michalek, "Fuzzy Metrics and Statistical Metric Spaces," *Kybernetika*, vol. 11, no. 5, pp. 336–334, 1975.
[3] K. Menger, "Statistical metrics," *Proceedings of the National Academy of Sciences pf the United States of America*, vol. 28, no. 12, pp. 535–537, 1942.

TABLE I

2D/3D/10D HYPERPLANES ESTIMATION ACCURACY AND NUMBER OF ITERATIONS OF THE REFINEMENT STAGE FOR (TOP) DIFFERENT OUTLIER RATIOS ω, (MIDDLE) DIFFERENT NOISE MAGNITUDES σ AND (BOTTOM) DIFFERENT SETTINGS FOR $\tau_I, \theta = \kappa \cdot \sigma$. WHEN KEPT CONSTANT: $\sigma = 1$, $\omega = 0.4$, $\kappa = 3$. LIGHTER BACKGROUND MEANS HIGHER PERFORMANCE.

2D

$\mu[\varepsilon]$ (°)						
ω	RANSAC	MSAC	FM-based RANSAC			
			$M^{1,1}$	$M^{1,2}$	$M^{2,1}$	$M^{2,2}$
0.60	4.43	3.14	4.63	4.10	3.97	3.10
0.50	3.03	2.33	2.61	2.33	2.28	1.88
0.40	2.13	1.81	1.77	1.59	1.57	1.33
0.20	1.58	1.53	0.96	0.88	0.89	0.80

$\mu[t]$					
ω	FM-based RANSAC				
	$M^{1,1}$	$M^{1,2}$	$M^{2,1}$	$M^{2,2}$	
0.60	9.11	10.44	10.79	11.05	
0.50	7.28	7.93	8.57	8.58	
0.40	6.47	6.71	7.68	7.40	
0.20	5.68	5.51	6.82	6.32	

$\mu[\varepsilon]$ (°)						
σ	RANSAC	MSAC	FM-based RANSAC			
			$M^{1,1}$	$M^{1,2}$	$M^{2,1}$	$M^{2,2}$
2.00	9.82	6.92	4.08	4.62	3.87	4.46
1.00	2.13	1.81	1.77	1.59	1.57	1.33
0.50	0.74	0.71	0.90	0.55	0.73	0.45
0.25	0.37	0.36	0.48	0.20	0.36	0.17

$\mu[t]$					
σ	FM-based RANSAC				
	$M^{1,1}$	$M^{1,2}$	$M^{2,1}$	$M^{2,2}$	
2.00	7.40	8.38	9.00	9.64	
1.00	6.47	6.71	7.68	7.40	
0.50	6.20	5.63	7.21	6.29	
0.25	6.02	4.93	6.87	5.57	

$\mu[\varepsilon]$ (°)						
κ	RANSAC	MSAC	FM-based RANSAC			
			$M^{1,1}$	$M^{1,2}$	$M^{2,1}$	$M^{2,2}$
4.00	2.85	2.09	1.85	1.84	1.65	1.54
3.00	2.13	1.81	1.77	1.59	1.57	1.33
2.50	2.03	1.88	1.71	1.45	1.52	1.23
2.00	2.18	2.18	1.65	1.29	1.47	1.13
1.00	3.60	3.58	1.47	1.04	1.35	1.01

$\mu[t]$					
κ	FM-based RANSAC				
	$M^{1,1}$	$M^{1,2}$	$M^{2,1}$	$M^{2,2}$	
4.00	6.05	6.26	7.14	6.81	
3.00	6.47	6.71	7.68	7.40	
2.50	6.79	7.02	8.03	7.94	
2.00	7.14	7.54	8.61	8.78	
1.00	8.61	10.65	10.66	13.56	

3D

$\mu[\varepsilon]$ (°)						
ω	RANSAC	MSAC	FM-based RANSAC			
			$M^{1,1}$	$M^{1,2}$	$M^{2,1}$	$M^{2,2}$
0.60	6.14	4.58	5.11	4.00	4.24	3.04
0.50	4.07	3.48	2.90	2.34	2.46	1.87
0.40	3.13	3.01	1.95	1.61	1.69	1.35
0.20	2.31	2.29	1.07	0.92	0.98	0.84

$\mu[t]$					
ω	FM-based RANSAC				
	$M^{1,1}$	$M^{1,2}$	$M^{2,1}$	$M^{2,2}$	
0.60	10.02	10.87	11.69	11.47	
0.50	7.55	8.10	9.05	8.83	
0.40	6.69	6.91	8.01	7.67	
0.20	5.73	5.55	6.87	6.41	

$\mu[\varepsilon]$ (°)						
σ	RANSAC	MSAC	FM-based RANSAC			
			$M^{1,1}$	$M^{1,2}$	$M^{2,1}$	$M^{2,2}$
2.00	13.32	9.97	5.21	4.92	4.53	4.32
1.00	3.13	3.01	1.95	1.61	1.69	1.35
0.50	1.11	1.08	0.99	0.57	0.79	0.47
0.25	0.64	0.63	0.52	0.21	0.39	0.18

$\mu[t]$					
σ	FM-based RANSAC				
	$M^{1,1}$	$M^{1,2}$	$M^{2,1}$	$M^{2,2}$	
2.00	8.68	9.63	10.45	10.66	
1.00	6.69	6.91	8.01	7.67	
0.50	6.32	5.82	7.43	6.52	
0.25	6.15	5.19	7.14	5.91	

$\mu[\varepsilon]$ (°)						
κ	RANSAC	MSAC	FM-based RANSAC			
			$M^{1,1}$	$M^{1,2}$	$M^{2,1}$	$M^{2,2}$
4.00	3.71	2.95	2.07	1.87	1.79	1.56
3.00	3.13	3.01	1.95	1.61	1.69	1.35
2.50	3.23	3.22	1.88	1.46	1.63	1.25
2.00	3.75	3.75	1.79	1.31	1.57	1.15
1.00	5.73	5.83	1.57	1.07	1.43	1.06

$\mu[t]$					
κ	FM-based RANSAC				
	$M^{1,1}$	$M^{1,2}$	$M^{2,1}$	$M^{2,2}$	
4.00	6.23	6.47	7.40	7.04	
3.00	6.69	6.91	8.01	7.67	
2.50	7.00	7.28	8.45	8.25	
2.00	7.40	7.87	9.04	9.17	
1.00	9.04	11.25	11.33	14.38	

10D

$\mu[\varepsilon]$ (°)						
ω	RANSAC	MSAC	FM-based RANSAC			
			$M^{1,1}$	$M^{1,2}$	$M^{2,1}$	$M^{2,2}$
0.60	10.69	10.18	7.46	4.19	5.53	3.00
0.50	8.82	8.77	3.56	2.26	2.83	1.78
0.40	6.92	6.94	2.29	1.54	1.88	1.29
0.20	5.66	5.70	1.17	0.90	1.03	0.84

$\mu[t]$					
ω	FM-based RANSAC				
	$M^{1,1}$	$M^{1,2}$	$M^{2,1}$	$M^{2,2}$	
0.60	14.12	14.53	15.61	14.52	
0.50	8.85	9.50	10.67	10.34	
0.40	7.30	7.60	8.88	8.50	
0.20	6.02	5.98	7.28	6.88	

$\mu[\varepsilon]$ (°)						
σ	RANSAC	MSAC	FM-based RANSAC			
			$M^{1,1}$	$M^{1,2}$	$M^{2,1}$	$M^{2,2}$
2.00	26.76	20.83	9.83	7.47	7.49	5.52
1.00	6.92	6.94	2.29	1.54	1.88	1.29
0.50	3.03	3.04	1.08	0.55	0.84	0.48
0.25	1.30	1.31	0.56	0.22	0.42	0.20

$\mu[t]$					
σ	FM-based RANSAC				
	$M^{1,1}$	$M^{1,2}$	$M^{2,1}$	$M^{2,2}$	
2.00	15.13	16.20	16.62	16.38	
1.00	7.30	7.60	8.88	8.50	
0.50	6.74	6.38	8.00	7.22	
0.25	6.45	5.81	7.56	6.63	

$\mu[\varepsilon]$ (°)						
κ	RANSAC	MSAC	FM-based RANSAC			
			$M^{1,1}$	$M^{1,2}$	$M^{2,1}$	$M^{2,2}$
4.00	6.26	6.09	2.50	1.85	2.03	1.49
3.00	6.92	6.94	2.29	1.54	1.88	1.29
2.50	7.75	7.86	2.16	1.39	1.80	1.20
2.00	9.03	9.13	2.03	1.26	1.71	1.14
1.00	12.61	12.61	1.71	1.10	1.52	1.12

$\mu[t]$					
κ	FM-based RANSAC				
	$M^{1,1}$	$M^{1,2}$	$M^{2,1}$	$M^{2,2}$	
4.00	6.96	7.16	8.14	7.74	
3.00	7.30	7.60	8.88	8.50	
2.50	7.78	8.03	9.28	9.11	
2.00	8.14	8.79	10.06	10.06	
1.00	10.06	12.41	12.58	15.87	

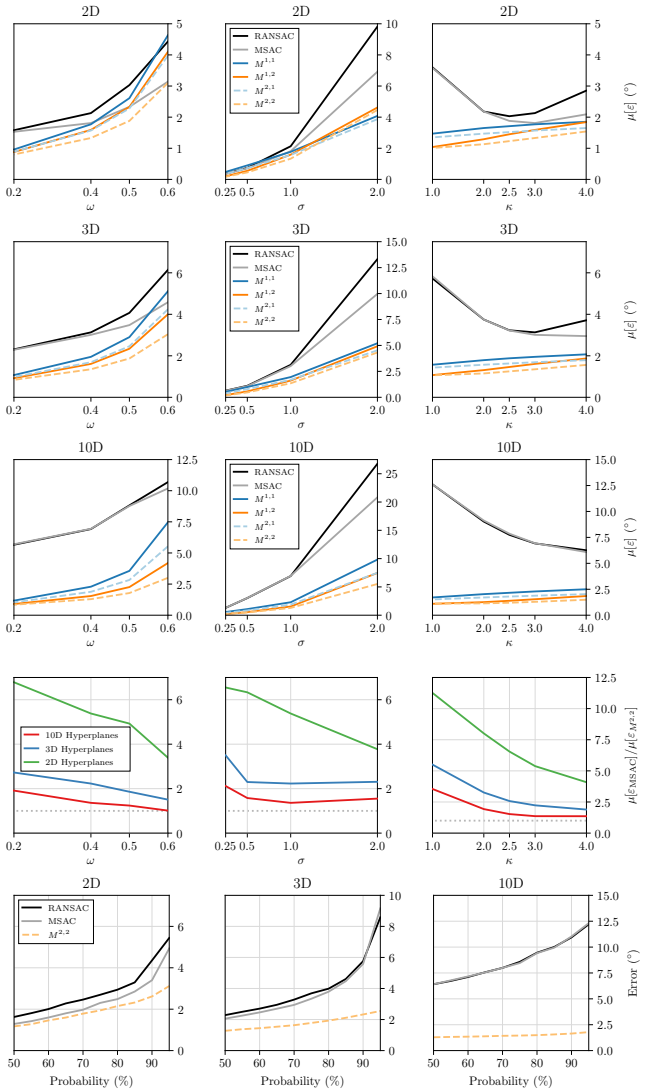


Fig. 1. (1st, 2nd, 3rd rows) Estimation accuracy for resp. 2D, 3D and 10D hyperplanes varying (left) the outlier ratio ω, (middle) the noise magnitude σ and (right) the setting of $\tau_I, \theta = \kappa \cdot \sigma$. (4th row) Ratio between the error of MSAC $\mu[\varepsilon_{\text{MSAC}}]$ and the error of FM-based RANSAC for $M^{2,2}$ $\mu[\varepsilon_{M^{2,2}}]$. (5th row) Comparison between FM-based RANSAC for $M^{2,2}$ and RANSAC/MSAC for different percentiles of the estimation error.

[4] A. George and P. Veeramani, "On some results in fuzzy metric spaces," *Fuzzy Sets and Systems*, vol. 64, no. 3, pp. 395–399, 1994.

[5] E. Klement, R. Mesiar, and E. Pap, *Triangular norms*. Springer, 2000.

[6] M. Grabiec, "Fixed points in fuzzy metric spaces," *Fuzzy Sets and Systems*, vol. 27, pp. 385–389, 1989.

[7] J. Camarena, V. Gregori, S. Morillas, and A. Sapena, "Fast detection and removal of impulsive noise using peer groups and fuzzy metrics," *Journal of Visual Communication and Image Representation*, vol. 19, no. 1, pp. 20–29, 2008.

[8] S. Morillas, V. Gregori, and G. Peris-Fajarnés, "New adaptative vector filter using fuzzy metrics," *Journal of Electronic Imaging*, vol. 16, no. 3, pp. 033 007:1–15, 2007.

[9] S. Morillas, V. Gregori, G. Peris-Fajarnés, and P. Latorre, "Isolating impulsive noise color images by peer group techniques," *Computer Vision and Image Understanding*, vol. 110, no. 1, pp. 102–116, 2008.

[10] S. Morillas, V. Gregori, G. Peris-Fajarnés, and A. Sapena, "Local self-adaptative fuzzy filter for impulsive noise removal in color image," *Signal Processing*, vol. 8, no. 2, pp. 390–398, 2008.

[11] V. Gregori, J. Miñana, and S. Morillas, "Some questions in fuzzy metric spaces," *Fuzzy Sets and Systems*, vol. 204, pp. 71–85, 2012.

[12] V. Gregori, S. Morillas, and A. Sapena, "On a class of completable fuzzy metric spaces," *Fuzzy Sets and Systems*, vol. 161, pp. 2193–2205, 2010.

[13] ——, "Examples of fuzzy metrics and applications," *Fuzzy Sets and Systems*, vol. 170, pp. 95–111, 2011.

[14] J. Dombi, "A general dass of fuzzy operators, the de morgan dass of fuzzy operators and fuzziness measures induced by fuzzy operators," *Fuzzy Sets and Systems*, vol. 8, pp. 149–163, 1982.

[15] P. J. Huber and E. M. Ronchetti, *Robust statistics*. Wiley, 2011.

[16] M. A. Fischler and R. C. Bolles, "Random sample consensus," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[17] O. Chum, J. Matas, and J. Kittler, "Locally Optimized RANSAC," in *Pattern Recognition (Proc. DAGM)*, ser. LNCS vol. 2781, B. Michaelis and G. Krell, Eds. Springer, 2003, pp. 236–243.

[18] A. Ortiz, E. Ortiz, J. J. Miñana, and O. Valero, "Hypothesis Scoring and Model Refinement Strategies for FM-based RANSAC," in *Proc. Spanish Conference on Fuzzy Logic and Technologies*, ser. Lecture Notes in Artificial Intelligence. Springer, 2021, *in press*.

[19] P. Torr and A. Zisserman, "MLESAC: A New Robust Estimator with Application to Estimating Image Geometry," *Computer Vision and Image Understanding*, vol. 78, no. 1, pp. 138–156, 2000.

Uso de t-normas para el estudio de la convexidad en conjuntos difusos intervalo-valuados

1st Pedro Huidobro
Dept. Statistics, O.R & M.E.
University of Oviedo
Oviedo, Spain
huidobropedro@uniovi.es

2nd Pedro Alonso
Dept. Mathematics
University of Oviedo
Oviedo, Spain
palonso@uniovi.es

3rd Humberto Bustince
Dept. of Statistics, Computing and Mathematics
University of Navarra
Navarra, Spain
bustince@unavarra.es

4th Vladimír Janiš
Dept. Mathematics
Matej Bel University
Banská Bystrica, Slovakia
vladimir.janis@umb.sk

5th Susana Montes
Dept. Statistics, O.R & M.E.
University of Oviedo
Oviedo, Spain
montes@uniovi.es

***Resumen*—En muchos problemas reales no se pueden tomar medidas de forma exacta. Así, los conjuntos difusos surgieron como una forma de intentar tratar con la incertidumbre de la forma más eficiente posible. Por otro lado, debe señalarse que la convexidad es un concepto interesante en varias áreas dentro de las matemáticas. Teniendo esto en cuenta, en este documento proponemos una extensión del concepto de convexidad para conjuntos difusos intervalo-valuados basada en el uso de t-normas para intervalos. Para ello, y teniendo en consideración la literatura científica existente respecto de t-normas, presentamos una definición de t-norma aplicada a intervalos. Por último, comprobamos que nuestra definición de convexidad, utilizando t-normas, preserva la convexidad a través de intersecciones, es decir, que la intersección de dos conjuntos difusos intervalo-valuados convexos es también convexa.**

***Index Terms*—Conjuntos difusos intervalo-valuados, t-normas, convexidad**

I. Introducción

En la vida real no todas las mediciones son exactas, por ello, en 1965, Zadeh [29] introdujo los conjuntos difusos para lidiar con la imprecisión. Desde entonces, muchos autores los han estudiado, así como a sus extensiones. Una de las más conocidas es la que da lugar a los conjuntos difusos intervalo-valuados, que fueron en la década de 1970 presentados independientemente por Zadeh [30], Grattan-Guiness [13], Jahn [16] y Sambuc [22]. Hoy en día, los estudios sobre este tipo de conjuntos han aumentado gracias a su aplicabilidad (ver [5], [7], [12], [22], [26]).

Por otro lado, la convexidad es una herramienta muy útil en muchos campos de las matemáticas (ver por ejemplo [17]–[19], [25]).

Este estudio ha sido parcialmente patrocinado por el programa español MINECO (TIN-2017-87600-P: P. Alonso; PGC2018-098623-B-I00: P. Huidobro and S. Montes), MICIN (PID2019!108392GB!I00: H. Bustince), la ayuda no. 1/0150/21 proporcionada por la agencia de subvenciones eslovaca VEGA (V. Janiš) y el programa de ayudas Severo Ochoa PA-20PF-BP19-169 (P. Huidobro).

Cuando Zadeh presentó los conjuntos difusos, estudió también su convexidad. Desde entonces, numerosos autores han trabajado sobre este tema, así como sobre la convexidad de sus extensiones, por ejemplo Ammar y Metz [1], Diaz et al. [8], Gupta y Dabgar [14], Ramik y Vlach [21], Sarkar [23], Syau y Lee [24], Yang [28] o Zhang et al. [31].

Según nuestro conocimiento, la convexidad en conjuntos difusos intervalo-valuados no ha sido estudiada en profundidad, por lo que Huidobro et al. [15] han trabajado recientemente en esa dirección.

Este trabajo está organizado del siguiente modo. Primero, en la sección 2, haremos una presentación de los conceptos necesarios para los desarrollos llevados a cabo. A continuación, en la sección 3 estudiaremos las t-normas para intervalos cerrados contenidos en el $[0,1]$. Finalmente, analizaremos si la convexidad de conjuntos difusos intervalo-valuados se preserva por intersecciones.

II. Conceptos básicos

Denotaremos por X al conjunto de referencia. Un conjunto difuso μ_A puede caracterizarse por una función de pertenencia $\mu_A : X \to [0,1]$. La colección de todos los conjuntos difusos sobre X se representa como $FS(X)$. Un conjunto difuso intervalo-valuado está caracterizado por una función $A : X \to L([0,1])$, donde $A(x) = [\underline{A(x)}, \overline{A(x)}]$ y $\underline{A(x)} \leq \overline{A(x)}$ para todo $x \in X$, y $L([0,1])$ son todos los intervalos cerrados contenidos en el $[0,1]$. Denotaremos como $IVFS(X)$ a la colección de todos los conjuntos difusos intervalo-valuados en X. Podemos observar que los conjuntos difusos intervalo-valuados son una generalización de los conjuntos difusos, donde la función de pertenencia toma como valor intervalos.

El primer problema que encontramos es cómo ordenar los elementos. Mientras que en $\mathbb{R}$ hay un orden natural ($a \leq b$ sii $b - a$ no es un número negativo), en $L([0,1])$ no ocurre lo mismo. Algo que parece lógico, a la hora de considerar órdenes, es respetar el orden reticular (*lattice order* en inglés)

[11], definido como $a \preceq_{Lo} b$ if $\underline{a} \leq \underline{b}$ y $\overline{a} \leq \overline{b}$ para cualquier $a = [\underline{a}, \overline{a}]$ y $b = [\underline{b}, \overline{b}]$ en $L([0,1])$. En este trabajo seguiremos las ideas de Bustince et al. [6] usando órdenes admisibles, que son órdenes totales refinando el order reticular. Para un estudio más detallado del uso de órdenes admisibles entre intervalos ver [15].

***Definición** 2.1:* [6] Sea $(L([0,1]), \preceq)$ un conjunto parcialmente ordenado. El orden $\preceq$ se dice admisible si verifica

i) $\preceq$ es un orden total en $L([0,1])$,
ii) para todo $[\underline{a}, \overline{a}], [\underline{b}, \overline{b}] \in L([0,1])$, $[\underline{a}, \overline{a}] \preceq [\underline{b}, \overline{b}]$ cuando $[\underline{a}, \overline{a}] \leq_{Lo} [\underline{b}, \overline{b}]$.

Los órdenes admisibles pueden generarse con funciones de agregación, así que recordemos su definición.

***Definición** 2.2:* [4], [20] Sea $\mathcal{A} : \bigcup_{i=1}^{n} [0,1]^n \to [0,1]$ cumpliendo

i) $\mathcal{A}(0,0,\ldots,0) = 0, \mathcal{A}(1,1,\ldots,1) = 1$,
ii) $\mathcal{A}$ es monótona en cada variable,

entonces $\mathcal{A}$ se llama función de agregación.

Hemos restringido el método propuesto por Bustince et al. [6] al intervalo $[0,1]$, ya que aunque el resultado original trabaja con funciones en $\mathbb{R}$, nosotros estamos centrados en el intervalo [0,1]:

***Proposición** 2.1:* Sean $\mathcal{A}, \mathcal{B} : [0,1]^2 \to [0,1]$ dos funciones de agregación continuas y sean $(u^*, v^*), (u', v') \in \{(u,v) \in [0,1]^2 \mid u \leq v\}$. Las igualdades $\mathcal{A}(u^*, v^*) = \mathcal{A}(u', v')$ y $\mathcal{B}(u^*, v^*) = \mathcal{B}(u', v')$ sólo pueden darse si $(u^*, v^*) = (u', v')$. Definida la relación $\preceq_{\mathcal{A},\mathcal{B}}$ en $L([0,1])$ por $a \preceq_{\mathcal{A},\mathcal{B}} b$ si y solo si

$$\mathcal{A}(\underline{a}, \overline{a}) < \mathcal{A}(\underline{b}, \overline{b})$$

o

$$\mathcal{A}(\underline{a}, \overline{a}) = \mathcal{A}(\underline{b}, \overline{b}) \text{ y } \mathcal{B}(\underline{a}, \overline{a}) \leq \mathcal{B}(\underline{b}, \overline{b})).$$

Entonces $\preceq_{\mathcal{A},\mathcal{B}}$ es un orden admisible en $L([0,1])$.

Con este método, no solo podemos construir nuevos órdenes admisibles, sino que también podemos comprobar que algunos de los órdenes más conocidos para intervalos son, en efecto, órdenes admisibles. Por ejemplo:

- Orden lexicográfico tipo 1 [6]: $a \preceq_{Lex1} b \Leftrightarrow \underline{a} < \underline{b}$ o $\underline{a} = \underline{b}$ y $\overline{a} \leq \overline{b}$, donde $\mathcal{A}(x,y) = x$ and $\mathcal{B}(x,y) = y$.
- Orden lexicográfico tipo 2 [6]: $a \preceq_{Lex2} b \Leftrightarrow \overline{a} < \overline{b}$ o $\overline{a} = \overline{b}$ y $\underline{a} \leq \underline{b}$, donde $\mathcal{A}(x,y) = y$ and $\mathcal{B}(x,y) = x$.
- Orden de Xu y Yager [27]: $a \preceq_{YX} b \Leftrightarrow \underline{a} + \overline{a} < \underline{b} + \overline{b}$ o $\underline{a} + \overline{a} = \underline{b} + \overline{b}$ y $\overline{a} - \underline{a} \leq \overline{b} - \underline{b}$, donde $\mathcal{A}(x,y) = x + y$ and $\mathcal{B}(x,y) = y - x$.

Por otro lado, dados dos conjuntos difusos μ_A y μ_B, se dice que μ_A está contenido en μ_B si $\mu_A(x) \leq \mu_B(x)$ [29]. Basándose en esta idea, si A y B son dos conjuntos difusos intervalo-valuados, decimos que A está o-contenido en B, y lo denotamos como $A \subseteq_o B$, si $A(x) \preceq_o B(x)$ para todo $x \in X$, donde $\preceq_o$ es una relación de orden en $L([0,1])$. Teniendo en cuenta este punto de vista, Huidobro et al. [15] definieron la intersección de conjuntos difusos intervalo-valuados.

***Definición** 2.3:* [15] Sean A, B dos conjuntos difusos intervalo-valuados en X y sea $\preceq_o$ una relación de orden en $L([0,1])$. Se llama o-intersección de A y B, $A \cap_o B$, al mayor conjunto difuso intervalo-valuado o-contenido en A y B.

En [15], Huidobro et al. probaron que esta definición depende del orden considerado en $L([0,1])$, obteniéndose distintas intersecciones para los distintos órdenes, pero con algo en común si se utilizan órdenes totales.

***Proposición** 2.2:* [15] Sea $\preceq_o$ un orden total en $L([0,1])$. Para cualquier $A, B \in IVFS(X)$, la o-intersección de A y B es el conjunto difuso intervalo-valuado definido por:

$$A \cap_o B(x) = \begin{cases} A(x) & \text{si } A(x) \preceq_o B(x), \\ B(x) & \text{si } B(x) \preceq_o A(x). \end{cases}$$

En consecuencia, resulta evidente que si A está o-incluido en B, entonces la intersección de A y B es A, ya que es el conjunto más grande contenido en ambos.

III. T-NORMAS PARA INTERVALOS

En esta sección repasaremos algunas de las definiciones que se pueden encontrar en la literatura sobre t-normas para después presentar nuestra propuesta.

Las t-normas en $[0,1]$ son funciones $t : [0,1] \times [0,1] \to [0,1]$ asociativas, conmutativas y crecientes en cada argumento verificando que $t(1,u) = u$ para todo $u \in [0,1]$. Para intervalos, Gehrke et al. [10] consideraron el orden reticular y el contenido usual. Partieron de que dado un punto $c \in L([0,1])$, tiene un intervalo asociado $[c,c]$ y con varias premisas más llegaron a la siguiente definición:

***Definición** 3.1:* [10] Una operación binaria T en $L([0,1])$ que es asociativa y conmutativa es una t-norma si para todo $a, b, c \in L([0,1])$ se cumplen las siguientes propiedades:

i) $T([u,u],[v,v])$ es un intervalo degenerado (origen y extremo coincidentes), donde $u, v \in [0,1]$,
ii) $T(a, b \vee c) = T(a,b) \vee T(a,c)$,
iii) $T(a, b \wedge c) = T(a,b) \wedge T(a,c)$,
iv) $T(a,[1,1]) = a$,
v) $T([0,1],[\underline{a}, \overline{a}]) = [0, \overline{a}]$,

donde $\vee$ y $\wedge$ se definen como:

$$[\underline{a}, \overline{a}] \vee [\underline{b}, \overline{b}] = [\underline{a} \vee \underline{b}, \overline{a} \vee \overline{b}],$$
$$[\underline{a}, \overline{a}] \wedge [\underline{b}, \overline{b}] = [\underline{a} \wedge \underline{b}, \overline{a} \wedge \overline{b}].$$

Además, también dieron una caracterización de las t-normas para intervalos.

***Teorema** 3.1:* [10] Toda t-norma en $L([0,1])$ se puede poner de la forma $T(a,b) = [t(\underline{a}, \underline{b}), t(\overline{a}, \overline{b})]$, donde t es una t-norma en $[0,1]$.

Además, fueron capaces de relajar la última condición de la definición si el operador binario es convexo, es decir, si $T(a,b) \leq_{Lo} c \leq_{Lo} T(d,e)$, entonces existen $r, s \in L([0,1])$ con $a \leq_{Lo} r \leq_{Lo} d$ y $b \leq_{Lo} s \leq_{Lo} e$ de manera que $c = T(r,s)$.

***Teorema** 3.2:* [10] Un operador binario convexo, conmutativo y asociativo, T en $L([0,1])$, es una t-norma si para todo $a, b, c \in L([0,1])$ se verifican las siguientes propiedades:

i) $T([u,u],[v,v])$ es un intervalo degenerado (origen y extremo coincidentes), donde $u, v \in [0,1]$,
ii) $T(a, b \vee c) = T(a,b) \vee T(a,c)$,

iii) $T(a, b \wedge c) = T(a, b) \wedge T(a, c)$,
iv) $T(a, [1, 1]) = a$,
v) $T([0, 1], [0, 1]) = [0, 1]$.

Bedregal y Takahashi [3] propusieron una extensión de la definición dada por Gehrke et al.:

***Definición** 3.2:* [3] Una aplicación T en $L([0, 1])$ es una t-norma para intervalos si para todo $a, b, c, d \in L([0, 1])$ se verifica:

i) $T(a, b) = T(b, a)$,
ii) $T(a, T(b, c)) = T(T(a, b), c))$,
iii) si $a \leq_{Lo} b$ y $c \leq_{Lo} d$, entonces $T(a, c) \leq_{Lo} T(b, d)$,
iv) si $a \subseteq b$ y $c \subseteq d$, entonces $T(a, c) \subseteq T(b, d)$,
v) $T(a, [1, 1]) = a$.

De este modo obtenemos t-normas para intervalos partiendo de t-normas.

***Proposición** 3.1:* [3] Sean t_1 y t_2 dos t-normas en $[0, 1]$. Si $t_1 \leq t_2$, entonces la aplicación $I[t_1, t_2]$, definida como $I[t_1, t_2](a, b) = [t_1(\underline{a}, \underline{b}), t_2(\overline{a}, \overline{b})]$, es una t-norma para intervalos llamada t-norma para intervalos derivada de t_1 y t_2.

Cuando consideramos un intervalo como un conjunto de números reales, tras hallar la intersección de a y b, $a \cap b$, tiene sentido que si $a_1 \subseteq a$ y $b_1 \subseteq b$, entonces $a_1 \cap b_1 \subseteq a \cap b$ [32]. Sin embargo, nosotros estamos considerando que un intervalo es una descripción imprecisa de la funcion de pertenencia, ya que estamos usando conjuntos difusos intervalo-valuados. Desde esta perspectiva, la monotonía en la inclusión es equivalente a ser creciente, esto es, si $A, B \in IVFS(X)$, $A \subseteq_o B$ si y solo si $A(x) \preceq_o B(x), \forall x \in X$.

Por ello, en este trabajo consideraremos la definición propuesta por Deschrijver [9]. En ella, elimina la cuarta condición de la Definición 3.2.

***Definición** 3.3:* [9] Una t-norma en $L([0, 1])$ es una función $T : L([0, 1]) \times L([0, 1]) \to L([0, 1])$ conmutativa, asociativa, creciente en ambos argumentos con respecto a $\leq_{Lo}$ y cumpliendo que $T([1, 1], a) = a$ para todo $a \in L([0, 1])$.

Partiendo de esta definición, podríamos definir t-norma para intervalos como:

***Definición** 3.4:* Sea $\preceq_o$ un orden en $L([0, 1])$. Una t-norma en $L([0, 1])$ es una función $T : L([0, 1]) \times L([0, 1]) \to L([0, 1])$ conmutativa, asociativa, creciente en ambos argumentos con respecto a $\leq_o$ y tal que $T([1, 1], a) = a$ para todo $a \in L([0, 1])$.

Una de las t-normas más conocidas es la función mínimo. En nuestro caso, el mínimo también es t-norma para intervalos.

***Proposición** 3.2:* Sea $\preceq_o$ un orden admisible en $L([0, 1])$. La aplicación $T_M : L([0, 1]) \times L([0, 1]) \to L([0, 1])$ definida como $T_M([\underline{a}, \overline{a}], [\underline{b}, \overline{b}]) = \min\{[\underline{a}, \overline{a}], [\underline{b}, \overline{b}]\}$ para cualquier $[\underline{a}, \overline{a}]$, $[\underline{b}, \overline{b}] \in L([0, 1])$ es una t-norma para intervalos.

Como hemos visto previamente, son interesantes aquellos casos donde podemos relacionar t-normas para intervalos con t-normas para números (ver [2], [3], [9]).

***Definición** 3.5:* Sea T una t-norma en $L([0, 1])$. T se dice representable si existen dos t-normas, t_1, t_2 tal que $T(a, b) = [t_1(\underline{a}, \underline{b}), t_2(\overline{a}, \overline{b})]$, para todo $a, b \in L([0, 1])$.

En general es más sencillo usar t-normas representables para intervalos, ya que simplifican la dificultad del problema.

***Proposición** 3.3:* Sean t_1, t_2 dos t-normas con $t_1 \leq t_2$. La función $T : L([0, 1]) \times L([0, 1]) \to L([0, 1])$ definida como $T(a, b) = [t_1(\underline{a}, \underline{b}), t_2(\overline{a}, \overline{b})]$ es una t-norma para intervalos con respecto al orden reticular.

***Nota** 3.1:* Este resultado no es cierto para cualquier orden. Si consideramos un ejemplo de orden admisible como es el orden lexicográfico tipo 1, podemos obtener que $[0{,}2, 0{,}7] \preceq_{Lex1} [0{,}3, 0{,}4]$. Sin embargo, considerando que t_1 y t_2 sean la t-norma mínimo, llegamos a que $T([0{,}1, 0{,}8], [0{,}2, 0{,}7]) = [0{,}1, 0{,}7]$ y $T([0{,}1, 0{,}8], [0{,}3, 0{,}4]) = [0{,}1, 0{,}4]$. Por tanto, $T([0{,}1, 0{,}8], [0{,}2, 0{,}7]) \not\preceq_{Lex1} T([0{,}1, 0{,}8], [0{,}3, 0{,}4])$, y podemos concluir que T no es creciente en la segunda componente.

Para otros órdenes admisibles pueden conseguirse contraejemplos similares.

En la Proposición 3.3 hemos obtenido un método para obtener t-normas aplicadas a intervalos a partir de t-normas, sin embargo, hemos visto en la Nota 3.1 que esto no siempre es posible. De hecho, hemos encontrado una caracterización en el siguiente resultado:

***Proposición** 3.4:* Sea $\preceq_o$ un orden total en $L([0, 1])$ y sean t_1 y t_2 dos t-normas en $[0, 1]$. Si consideramos la aplicación T definida como $T(a, b) = [t_1(\underline{a}, \underline{b}), t_2(\overline{a}, \overline{b})]$, entonces tenemos que T es una t-norma para intervalos si y solo si T es creciente en el segundo argumento.

IV. Convexidad de conjuntos difusos intervalo-valuados utilizando t-normas

En este trabajo hemos considerado la definición de convexidad propuesta por Huidobro et al. [15]:

***Definición** 4.1:* Sea $(X, \leq)$ un espacio ordenado y sea $\preceq_o$ una relación de orden en $L([0, 1])$. Un conjunto difuso intervalo-valuado A en X se dice o-convexo, si para cada $x \leq y \leq z$ en X se cumple $A(x) \preceq_o A(y)$ o $A(z) \preceq_o A(y)$.

Claramente esta definición generaliza la idea de convexidad en conjuntos difusos propuesta por Zadeh [29], ya que si consideramos un conjunto difuso μ_A como conjunto difuso intervalo-valuado, esto es, $A(x) = [\mu_A(x), \mu_A(x)], \forall x \in X$, diremos que A es convexo si se cumple $A(x) \preceq_o A(y)$ o $A(z) \preceq_o A(y)$. Esto es equivalente a decir que $[\mu_A(x), \mu_A(x)] \preceq_o [\mu_A(y), \mu_A(y)]$ o $[\mu_A(z), \mu_A(z)] \preceq_o [\mu_A(y), \mu_A(y)]$, que con el uso de cualquier orden que refine al orden reticular, por ejemplo cualquiera de los órdenes admisibles, equivale a $\mu_A(x) \leq \mu_A(y)$ o $\mu_A(z) \leq \mu_A(y)$. Con esto recuperamos la definición de convexidad para conjuntos difusos propuesta por Zadeh, que venía dada por: μ_A es convexo si y solo si $\mu_A(y) \geq \min\{\mu_A(x), \mu_A(z)\}$ para $x < y < z$.

Cuando trabajamos con órdenes totales, esta definición es equivalente a $\min\{A(x), A(z)\} \preceq_o A(y)$, y nosotros hemos podido comprobar satisfactoriamente que el mínimo es una t-norma para intervalos. Partiendo de esa idea, reemplazaremos el mínimo por otra t-norma para intervalos. Por lo que proponemos la siguiente definición:

***Definición** 4.2:* Sea $(X, \leq)$ un espacio ordenado, sea $\preceq_o$ una relación de orden total en $L([0, 1])$ y sea T una t-norma para intervalos. Un conjunto difuso intervalo-valuado A en X

se dice T-o-convexo, si para cada $x \leq y \leq z$ en X se cumple $T(A(x), A(z)) \preceq_o A(y)$.

Comprobemos la validez de la definición comprobando su compatibilidad con la intersección.

***Proposición** 4.1:* Sea $(X, \leq)$ un espacio ordenado, sea $\preceq_o$ una relación de orden total en $L([0,1])$ y sea T una t-norma para intervalos. La intersección de dos conjuntos difusos intervalo-valuados T-o-convexos es T-o-convexa.

Este resultado apoya nuestra definición de t-normas para intervalos mientras que las otras consideradas no conservaban esta propiedad. Veamos un ejemplo.

***Ejemplo** 4.1:* Consideremos el orden lexicográfico tipo 1 y la Definición 3.1. Según esta definición, el operador $T^*(a,b) = [\text{mín}\{\underline{a},\underline{b}\}, \text{mín}\{\overline{a},\overline{b}\}]$ es una t-norma en $L([0,1])$. Consideremos $X = \{x,y,z\}$ con $x < y < z$ y los conjuntos difusos intervalo-valuados A y B definidos por $A(x) = [0{,}3, 1]$, $A(y) = [0{,}3, 0{,}8]$ y $A(z) = [0{,}7, 0{,}8]$, y $B(x) = [0{,}4, 0{,}5]$, $B(y) = [0{,}5, 0{,}7]$ y $B(z) = [0{,}6, 0{,}9]$.

La intersección resulta ser $(A \cap_{Lex1} B)(x) = [0{,}3, 1]$, $(A \cap_{Lex1} B)(y) = [0{,}3, 0{,}8]$ y $(A \cap_{Lex1} B)(z) = [0{,}6, 0{,}9]$. Es fácil comprobar que A y B son T^*-$Lex1$-convexos, pero $A \cap_{Lex1} B$ no, ya que $T^*(A \cap_{Lex1} B(x), A \cap_{Lex1} B(z)) = T^*([0{,}3, 1], [0{,}6, 0{,}9]) = [0{,}3, 0{,}9] \not\preceq_{Lex1} [0{,}3, 0{,}8] = A \cap_{Lex1} B(y)$. Este contraejemplo es válido también para la Definición 3.2.

V. Conclusiones

En este trabajo hemos presentado una definición de t-norma para intervalos. Utilizando este concepto, hemos generalizado la noción de convexidad para conjuntos difusos intervalo-valuados y hemos podido comprobar que la definición propuesta es compatible con la intersección de conjuntos difusos intervalo-valuados.

Referencias

[1] E. Ammar and J. Metz. On fuzzy convexity and parametric fuzzy optimization. *Fuzzy Sets and Systems*, 49(2):135 – 141, 1992.

[2] B. Bedregal and A. Takahashi. Interval t-norms as interval representations of t-norms. In *The 14th IEEE International Conference on Fuzzy Systems, 2005. FUZZ '05.*, pages 909–914, 2005.

[3] B. Bedregal and A. Takahashi. The best interval representations of t-norms and automorphisms. *FuzzySetsandSystems*, 157:3220–3230, 2006.

[4] G. Beliakov, H. Bustince, and T. Calvo. *A Practical Guide to Averaging Functions*. Springer, 2016.

[5] H. Bustince and P. Burillo. Mathematical analysis of interval-valued fuzzy relations: Application to approximate reasoning. *Fuzzy Sets and Systems*, 113(2):205 – 219, 2000.

[6] H. Bustince, J. Fernandez, A. Kolesárová, and R. Mesiar. Generation of linear orders for intervals by means of aggregation functions. *Fuzzy Sets and Systems*, 220:69 – 77, 2013.

[7] C. Cornelis, G. Deschrijver, and E.E. Kerre. Implication in intuitionistic fuzzy and interval-valued fuzzy set theory. *International Journal of Approximate Reasoning*, 35:55 – 95, 2004.

[8] S. Díaz, E. Induráin, V. Janiš, J. V. Llinares, and S. Montes. Generalized convexities related to aggregation operators of fuzzy sets. *Kybernetika*, 53:383 – 393, 2017.

[9] G. Deschrijver. Arithmetic operators in interval-valued fuzzy set theory. *Information Sciences*, 177(14):2906–2924, 2007.

[10] M. Gehrke, C. Walker, and E. Walker. Some comments on interval valued fuzzy sets. *International Journal of Intelligent Systems*, 11(10):751–759, 1996.

[11] J.A Goguen. L-fuzzy sets. *Journal of Mathematical Analysis and Applications*, 18(1):145 – 174, 1967.

[12] M.B. Gorzalczany. A method of inference in approximate reasoning based on interval-valued fuzzy sets. *Fuzzy Sets and Systems*, 21:1 – 17, 1987.

[13] I. Grattan-Guinness. Fuzzy membership mapped onto intervals and many-valued quantities. *Mathematical Logic Quarterly*, 22(1):149 – 160, 1976.

[14] S.K. Gupta and D. Dangar. Duality for a class of fuzzy nonlinear optimization problem under generalized convexity. *Fuzzy Optimization and Decision Making*, 13:131 – 150, 2014.

[15] P. Huidobro, P. Alonso, V. Janiš, and S. Montes. Orders preserving convexity under intersections for interval-valued fuzzy sets. In Marie-Jeanne Lesot, Susana Vieira, Marek Z. Reformat, João Paulo Carvalho, Anna Wilbik, Bernadette Bouchon-Meunier, and Ronald R. Yager, editors, *Information Processing and Management of Uncertainty in Knowledge-Based Systems*, pages 493 – 505, Cham, 2020. Springer International Publishing.

[16] K.U. Jahn. Intervall-wertige mengen. *Mathematische Nachrichten*, 68:115 – 132, 1975.

[17] Y. Kim, Z. Xi, and J. Lien. Disjoint convex shell and its applications in mesh unfolding. *Computer-Aided Design*, 90:180 – 190, 2017.

[18] M. Li, W. Yang, and X. Zhang. Projection on convex set and its application in testing force closure properties of robotic grasping. In *Intelligent Robotics and Applications*, volume 6425, pages 240 – 251. Springer, 11 2010.

[19] L. Liberti. Reformulation and convex relaxation techniques for global optimization. *Quarterly Journal of the Belgian, French and Italian Operations Research Societies*, 2(3):255 – 258, 2004.

[20] R. Mesiar and M. Komorníková. Aggregation functions on bounded posets. In Chris Cornelis, Glad Deschrijver, Mike Nachtegael, Steven Schockaert, and Yun Shi, editors, *35 Years of Fuzzy Set Theory: Celebratory Volume Dedicated to the Retirement of Etienne E. Kerre*, pages 3 – 17, Berlin, Heidelberg, 2011. Springer.

[21] J. Ramik and M Vlach. *Generalized Concavity in Fuzzy Optimization and Decision Analysis*. Kluwer Academic Publishers, 2002.

[22] R. Sambuc. Function phi-flous, application a l'aide au diagnostic en pathologie thyroidienne. *These de Doctorat en Medicine*, 1975.

[23] D. Sarkar. Concavoconvex fuzzy set. *Fuzzy Sets and Systems*, 79:267 – 269, 1996.

[24] Y. Syau and E Lee. Fuzzy convexity and multiobjective convex optimization problems. *Computers & Mathematics with Applications*, 52(3):351 – 362, 2006.

[25] M. Tofighi, O. Yorulmaz, K. Kose, D.C. Kahraman, R. Cetin-Atalay, and A.E. Cetin. Phase and tv based convex sets for blind deconvolution of microscopic images. *IEEE Journal of Selected Topics in Signal Processing*, 10(1):81 – 91, 2015.

[26] I.B. Turksen and Z. Zhong. An approximate analogical reasoning schema based on similarity measures and interval-valued fuzzy sets. *Fuzzy Sets and Systems*, 34:323 – 346, 1990.

[27] Z.S. Xu and R.R. Yager. Some geometric aggregation operators based on intuitionistic fuzzy sets. *International Journal of General Systems*, 35(4):417 – 433, 2006.

[28] X. Yang. A property on convex fuzzy sets. *Fuzzy Sets and Systems*, 126:269 – 271, 2002.

[29] L.A. Zadeh. Fuzzy sets. *Information and Control*, 8(3):338 – 353, 1965.

[30] L.A. Zadeh. The concept of a linguistic variable and its application to approximate reasoning—I. *Information Sciences*, 8(3):199 – 249, 1975.

[31] J. Zhang, Q. Zheng, X. Ma, and L. Li. Relationships between interval-valued vector optimization problems and vector variational inequalities. *Fuzzy Optimization and Decision Making*, 15:33 – 55, 2016.

[32] Q. Zuo. Description of strictly monotonic interval and/or operations. *Reliable Computing*, pages 23 – 25, 1995.

XX Congreso Español sobre Tecnologías y Lógica Fuzzy

Sesión especial:

Toma de decisiones con información difusa



Preferencias no lineales en Toma de Decisión en Grupo. Amplificación de Valores Extremos

1st Diego García-Zamora
Departamento de Informática
Universidad de Jaén
Jaén, España
dgzamora@ujaen.es

2nd Álvaro Labella
Departamento de Informática
Universidad de Jaén
Jaén, España
alabella@ujaen.es

3rd Rosa M. Rodríguez
Departamento de Informática
Universidad de Jaén
Jaén, España
rmrodrig@ujaen.es

4st Luis Martínez
Departamento de Informática
Universidad de Jaén
Jaén, España
martin@ujaen.es

***Resumen*—Varios estudios han demostrado que el uso de escalas no lineales mejoran las decisiones obtenidas en problemas de Toma de Decisión en Grupo (TDG). Este trabajo está orientado a incorporar estas escalas no lineales en Procesos de Alcance de Consenso (PAC), los cuales son la herramienta fundamental para suavizar los conflictos que aparecen en los problemas de TDG. Para ello, utilizaremos automorfismos no lineales definidos en el intervalo unidad para deformar las preferencias de los expertos, expresadas mediante relaciones de preferencia difusas, con el objetivo de obtener escalas más realistas. Este trabajo introduce estas deformaciones no lineales como Amplificaciones de Valores Extremos (AVEs), analiza sus principales propiedades y presenta dos familias concretas de AVEs: una basada en la función seno y otra basada en polinomios. Por último estudiamos mediante un caso práctico cómo influye este enfoque no lineal basado en AVEs en dos modelos de consenso clásicos para TDG.**

***Palabras clave*—Toma de Decisión en Grupo, Procesos de Alcance de Consenso, Amplificación de Valores Extremos, Preferencias no lineales.**

I. Introducción

Tradicionalmente se han utilizado escalas lineales para modelar las preferencias en los problemas de Toma de Decisión en Grupo (TDG). Sin embargo, estudios recientes han probado que se obtienen mejores decisiones usando escalas no lineales para representar las preferencias de los expertos [1], [2]. Sin embargo, ninguna de esas propuestas tiene en cuenta los conflictos entre expertos que normalmente aparecen en los problemas de TDG.

En este trabajo estudiaremos el efecto de deformar las preferencias de los expertos mediante escalas no lineales en Procesos de Alcance de Consenso (PACs) para problemas de TDG. Asumiremos que las preferencias de los expertos vienen dadas por medio de Relaciones de Preferencia Difusas (RPDs) y aplicaremos una deformación no lineal a cada una de estas preferencias para *ajustar* los valores iniciales a una escala más realista. Por último, estudiaremos el impacto de este enfoque no lineal en los modelos de consenso propuestos en [3], [4] analizando el grado de consenso obtenido y número de rondas empleado.

Esta contribución está dividida como sigue: En la Sección II se revisan brevemente los problemas de TDG y los PAC. La Sección III está dedicada a la noción principal de este trabajo, la Amplificación de Valores Extremos (AVE), definidas como aquellos automorfismos del intervalo $[0,1]$ que incrementan la distancia entre valores extremos de las preferencias dadas mediante RPDs. En la Sección IV se muestra cómo influyen estas escalas no lineales en los PAC. Finalmente, la Sección V concluye el trabajo.

II. Toma de Decisión en Grupo y Procesos de Alcance de Consenso

En esta sección se revisan brevemente los principales conceptos relativos a TDG y PACs.

Un problema de TDG es una situación en la que un grupo de expertos $E = \{e_1, e_2, ..., e_m\}$, $2 \leq m \in \mathbb{N}$, tiene que elegir la mejor solución dentro de un conjunto de posibles alternativas $X = \{X_1, X_2, ..., X_n\}$, $2 \leq n \in \mathbb{N}$. Sin pérdida de generalidad, podemos suponer que las opinión de cada experto viene dada por medio de una RPD, la cual consiste en una matriz $P_k \in \mathcal{M}_{n\times n}([0,1])$ donde cada entrada $p_{ij}^k \in [0,1]$ representa el grado en el que el experto e_k prefiere la alternativa X_i sobre la X_j. Las RPDs verifican la condición de simetría $p_{ij}^k + p_{ji}^k = 1 \ \forall \ i,j \in \{1,2,...,n\}, k \in \{1,2,...,m\}$, conocida como reciprocidad aditiva.

Es posible que en el proceso de resolución de un problema de TDG surjan conflictos entre las opiniones de los expertos y que algunos consideren que sus opiniones no se han tenido suficientemente en cuenta durante el proceso [5], [6]. En estos casos se aplican PACs para lograr un acuerdo en la solución elegida [7], [8]. Un PAC es un proceso iterativo que usa una medida de consenso para calcular la cercanía entre las preferencias de los expertos [6] y finaliza cuando se alcanza un grado de consenso predefinido o se alcanza un determinado número de rondas. En este trabajo usaremos los modelos de consenso propuestos en [3], [4], ya que han demostrado un buen funcionamiento [8].

III. Amplificación de Valores Extremos

En esta sección se introduce la definición de AVE, así como algunos ejemplos concretos de familias de funciones que cumplen con los requisitos que caracterizan a estas AVEs.

Definición 1 (Amplificación de Valores Extremos)**.** *Llamaremos Amplificación de Valores Extremos (AVE) a una función* $D : [0,1] \to [0,1]$ *verificando:*

1. *D es un automorfismo en el intervalo* $[0,1]$,

2. *D es una función de clase $\mathcal{C}^1$, esto es, D es derivable en $[0,1]$ y su derivada D' es continua en $[0,1]$,*
3. *D verifica $D(x) = 1 - D(1-x)\ \forall\ x \in [0,1]$,*
4. *$D'(0) > 1$ y $D'(1) > 1$,*
5. *D es cóncava en un entorno de 0 y convexa en un entorno de 1.*

El propósito de las AVEs es transformar las preferencias de un problema de TDG de forma que los nuevos valores obedezcan una escala no lineal en la que las distancias entre los valores más extremos se ven incrementadas respecto a las diferencias originales.

Nota 1. *Aunque el propósito y las características de las AVEs están orientadas al modelado de las escalas no lineales para las preferencias en PAC, el hecho de que permitan definir medidas de similaridad junto al enfoque particular que proporcionan a la hora de tratar la información más extrema recuerda al Hypermatching introducido por Yager y Petry [14]. En el futuro nos gustaría estudiar detalladamente cómo relacionar ambas propuestas.*

La primera propiedad, junto con las condiciones de regularidad de la segunda, garantizan que la biyectividad de las AVEs. Esto es fundamental al comparar preferencias, puesto que no deseamos que dos valores diferentes de las preferencias originales tengan por imagen el mismo valor. La tercera propiedad está orientada a asegurar la reciprocidad aditiva de la matriz obtenida al aplicar la AVE a cada uno de los items de una RPD. La cuarta propiedad está relacionada con la amplificación de las distancias entre los valores extremos de las preferencias originales (los cercanos a 0 y a 1), mientras que la quinta permite deducir que a medida que nos acercamos a los extremos, la amplificación de las distancias será mayor. Formalmente:

Teorema 1. *Sea $D : [0,1] \to [0,1]$ un AVE. Entonces*

1. *La función $d_D : [0,1] \times [0,1] \to [0,1]$ definida por*

$$d_D(x,y) = |D(x) - D(y)|\ \forall\ x, y \in [0,1],$$

es una Disimilitud Restringida [9] y la función $S_D : [0,1] \times [0,1] \to [0,1]$ definida por

$$S_D(x,y) = 1 - |D(x) - D(y)|\ \forall\ x, y \in [0,1].$$

es una Función de Equivalencia Restringida [9],

2. *Podemos encontrar tres intervalos $I_1,\ I_2,\ I_3 \subset [0,1]$ tales que $0 \in I_1,\ 1 \in I_3$, y $I_1 < I_2 < I_3$ verificando*

$$|D(y) - D(x)| > |y - x|\ \forall\ x, y \in I_1\ :\ x \neq y,$$
$$|D(y) - D(x)| < |y - x|\ \forall\ x, y \in I_2\ :\ x \neq y,$$
$$|D(y) - D(x)| > |y - x|\ \forall\ x, y \in I_3\ :\ x \neq y,$$

3. *El gráfico de D está por encima de la diagonal del cuadrado $[0,1] \times [0,1]$ para valores cercanos a 0 y está bajo la misma diagonal para los valores cercanos a 1,*
4. *Existen un entorno U_0 de 0 y un entorno U_1 de 1 tales que para cada par $x, y \in U_0^\circ,\ x < y$ existe $h_0 > 0$ tal que $|D(x) - D(x-t)| \geq |D(y) - D(y-t)|\ \ \forall\ t \in [0, h_0]$ y para cada par $x, y \in U_1^\circ,\ x < y$, existe $h_1 > 0$ tal que $|D(x-t) - D(x)| \leq |D(y-t) - D(y)|\ \ \forall\ t \in [0, h_1]$.*

Demostración. La prueba de este resultado es consecuencia del Teorema del Valor Medio. Omitimos los detalles concretos por limitación de espacio. □

Las cuatro tesis de este resultado permiten interpretar las características de las AVEs de la siguiente manera:

1. Transforman RPDs en RPDs.
2. Amplifican las distancias entre los valores extremos y reducen la distancia entre los valores intermedios.
3. Su gráfica tiene un patrón común (ver Figura 1).
4. La amplificación de las distancias se incrementa a medida que nos acercamos a los extremos.

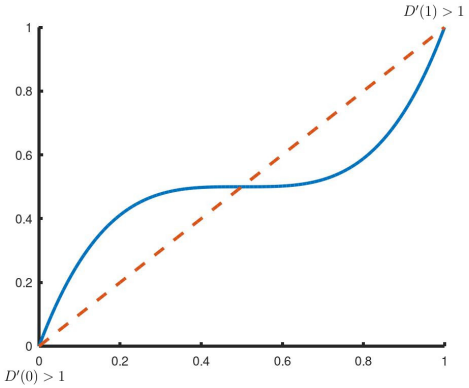


Figura 1: Esbozo del gráfico de una AVE

A continuación presentamos un par de familias de AVEs.

AVEs basadas en la función seno.

Sea $\alpha \in]0, \frac{1}{2\pi}]$. La función de clase $\mathcal{C}^\infty$ $s_\alpha : [0,1] \to [0,1]$ dada por

$$s_\alpha(x) = x - \alpha \cdot \sin(2\pi x - \pi)\ \forall\ x \in [0,1]$$

es una AVE (ver la Figura 2).

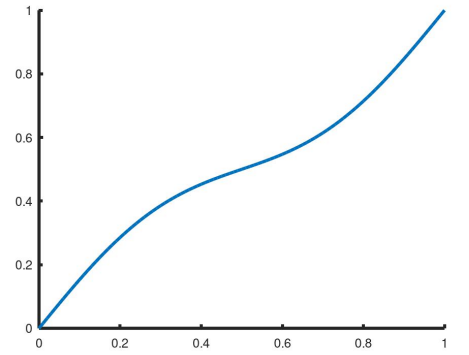


Figura 2: AVE $s_{0,09}$

AVEs basadas en polinomios.

Fijado $\alpha > 1$, la función $m_\alpha : [0,1] \to [0,1]$ dada por

$$m_\alpha(x) = \begin{cases} \frac{1}{2} - \frac{1}{2}(1-2x)^\alpha & 0 \leq x < \frac{1}{2} \\ \frac{1}{2} + \frac{1}{2}(2x-1)^\alpha & \frac{1}{2} \leq x \leq 1 \end{cases},$$

es una AVE (ver las Figuras 3 y 4).

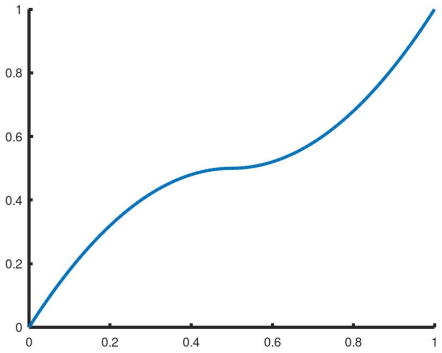


Figura 3: AVE m_2.

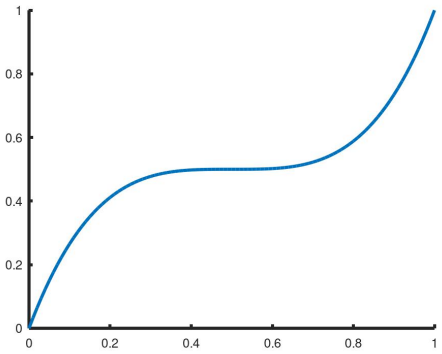


Figura 4: AVE $m_{3,39}$.

IV. Amplificación de Valores Extremos en Procesos de Alcance de Consenso

En esta sección mostramos el funcionamiento de las AVEs cuando son utilizadas en PAC para TDG. Para ello, hemos implementado el enfoque no lineal de las AVEs en los PAC ampliamente utilizados en la literatura y presentados en [3], [4], que están implementados en el software AFRYCA [6].

El problema de TDG considerado consiste en grupo de 100 expertos que enuncian sus preferencias sobre las alternativas $X = \{X_1, X_2, X_3, X_4\}$ por medio de RPDs (ver [10]). Para hacer las simulaciones hemos usado los valores por defecto que AFRYCA define para los parámetros de cada modelo. El umbral de consenso deseado se ha establecido en 0.85 y el máximo número de rondas permitido es 15. Los resultados obtenidos se muestran en las Tablas I y II.

AFRYCA [6] proporciona una herramienta para visualizar los resultados de las distintas simulaciones mediante la técnica de escalado multidimensional [11] (ver Figuras 5 y 6). Esta representación muestra la opinión colectiva de los expertos en el centro del gráfico y a su alrededor las preferencias de los expertos.

Tabla I: Resultados con Herrera-Viedma et al. [3]

AVE	*Orden de alternativas*	*Rondas*	*Consenso*
Clásico	$x1 \succ x2 \succ x4 \succ x3$	6	0.87
$s_{0,08}$	$x1 \succ x2 \succ x4 \succ x3$	6	0.89
$s_{0,09}$	$x1 \succ x2 \succ x4 \succ x3$	6	0.92
m_2	$x1 \succ x2 \succ x4 \succ x3$	5	0.86
$m_{3,39}$	$x1 \succ x2 \succ x4 \succ x3$	5	0.91

Tabla II: Resultados con Quesada et al. [4]

AVE	*Orden de alternativas*	*Rondas*	*Consenso*
Clásico	$x4 \succ x1 \sim x2 \sim x3$	10	0.85
$s_{0,08}$	$x4 \succ x1 \succ x2 \sim x3$	7	0.86
$s_{0,09}$	$x4 \succ x1 \succ x2 \sim x3$	7	0.87
m_2	$x4 \succ x1 \succ x2 \sim x3$	7	0.86
$m_{3,39}$	$x4 \succ x1 \succ x2 \sim x3$	5	0.87

El modelo clásico [3] alcanzó un grado de consenso de $0{,}87$ en 6 rondas. Para este modelo, las AVEs $s_{0,08}$ y $s_{0,09}$ han mejorado el grado de consenso alcanzado, aunque no han reducido el número de rondas. Las AVEs polinómicas m_2 y $m_{3,39}$ han reducido el número de rondas, obteniendo grados de consenso de $0{,}86$ y $0{,}91$ respectivamente en 5 rondas.

Por otro lado, el modelo clásico [4] necesitó 10 rondas para lograr un grado de consenso de $0{,}85$. En este caso, ambas familias de AVEs han mejorado significativamente el modelo original. Las AVEs $s_{0,08}$, $s_{0,09}$ y m_2 han mejorado ligeramente el grado de consenso en 7 rondas, mientras que la AVE $m_{3,39}$ destaca por haber incrementado el grado de consenso en sólo 5 rondas. Las simulaciones indican que las AVEs mejoran ambos modelos ya que o bien disminuyen el número de rondas o bien incrementan el grado de consenso alcanzado.

Para analizar el comportamiento de las AVEs en un PAC desde un punto de vista teórico es necesario tener en cuenta dos aspectos clave. Por un lado, es un hecho probado que en los modelos de consenso los valores menos extremos de las preferencias aportan cohesión al grupo y favorecen el alcance del consenso colectivo [12], [13]. Por otro lado, si bien es cierto que las AVEs amplifican la distancia entre los valores extremos, cuando usamos una AVE los valores intermedios se acercan entre sí. Considerando estos dos hechos conjuntamente podemos deducir que al aplicar una AVE a las preferencias que van a ser usadas en un modelo de consenso, el modelo de consenso modificado por la AVE alcanzará generalmente el grado de consenso deseado más rápido que el modelo original puesto que los valores intermedios de las preferencias, esto es, la información *más importante* para el consenso, están *más cerca* desde el principio.

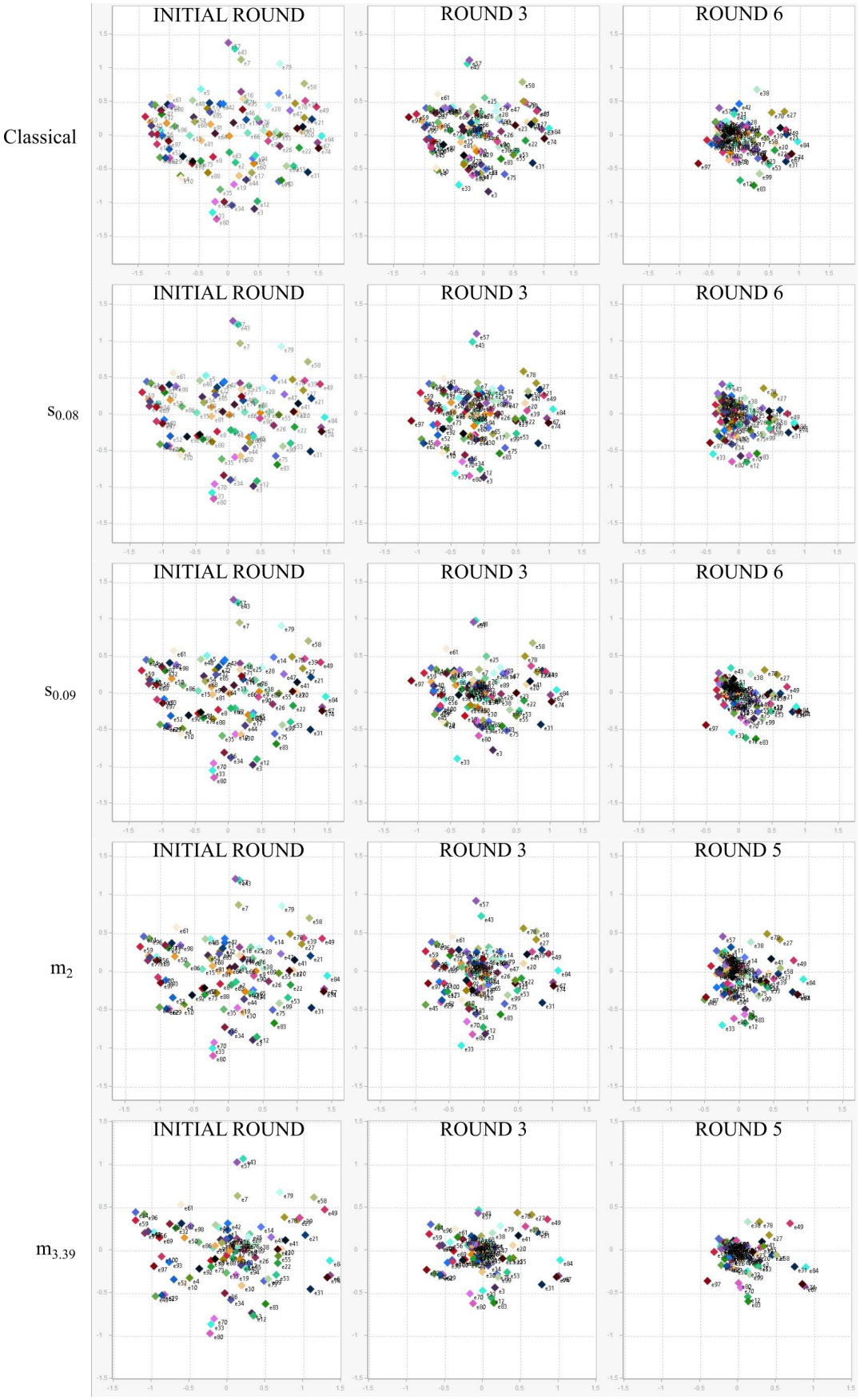


Figura 5: Herrera Viedma et al. [3] simulations.



Figura 6: Quesada et al. [4] simulations.

Para que los resultados obtenidos sean fiables, es imprescindible tener en cuenta que aunque la AVE usada en el PAC modifique los valores iniciales de las preferencias, el orden de preferencia de las alternativas en cada RPD no sufre alteraciones significativas. Omitimos aquí los detalles concretos por motivos de espacio, pero es posible demostrar que las familias de AVEs propuestas en este trabajo, s_α, y m_α no introducen cambios significativos en el orden de las alternativas preferidas por cada experto.

V. Conclusiones

Los PAC clásicos asumen escalas lineales para los valores de preferencias de los expertos. Este trabajo propone el uso de escalas no lineales en estos modelos con el objetivo de modelar las preferencias de los expertos de una forma más realista.

Este trabajo presenta las propiedades analíticas de estas escalas no lineales y estudia sus principales propiedades. Estas deformaciones no lineales han recibido el nombre de AVEs y se caracterizan por transformar RPDs en RPDs de forma que las distancias entre los valores extremos se incrementan mientras que las distancias entre los valores intermedios se reducen. Para mostrar el impacto de las AVEs en los PAC, hemos utilizado el software AFRYCA para simular el funcionamiento de dos modelos de consenso cuando se combinan con las AVEs definidas en este trabajo. Los resultados muestran que o bien se reducen el número de rondas necesarias para alcanzar el consenso o bien se incrementa el grado de consenso.

Como trabajos futuros definiremos nuevas AVEs e intentaremos optimizar los parámetros de las AVEs existentes para modelos de consenso concretos de acuerdo con diferentes métricas. También pretendemos analizar el comportamiento de las AVEs en otros modelos de consenso, así como aplicar este marco teórico para resolver problemas de decisión del mundo real.

Agradecimientos

Este trabajo ha sido parcialmente subvencionado por el Ministerio de Economía y Competitividad mediante el proyecto PGC2018-099402-B-I00, y la Ayuda Postdoctoral Ramón y Cajal (RYC-2017-21978), y por el Ministerio de Ciencia, Innovación y Universidades mediante la Ayuda Predoctoral para la Formación de Profesorado Universitario (FPU2019/01203).

Referencias

[1] J. Masthoff, "Group modeling: Selecting a sequence of television items to suit a group of viewers," *User Model. User-Adapt. Interact.*, vol. 14, pp. 37–85, 02 2004.

[2] A. Delic, F. Ricci, and J. Neidhardt, "Preference networks and non-linear preferences in group recommendations," in *IEEE/WIC/ACM International Conference on Web Intelligence*, 10 2019, pp. 403–407.

[3] E. Herrera-Viedma, F. Herrera, and F. Chiclana, "A consensus model for multiperson decision making with different preference structures," *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, vol. 32, pp. 394 – 402, 06 2002.

[4] F. J. Quesada, I. Palomares, and L. Martínez, "Managing experts behavior in large-scale consensus reaching processes with uninorm aggregation operators," *Applied Soft Computing*, vol. 35, pp. 873–887, 2015.

[5] Á. Labella, Y. Liu, R. M. Rodríguez, and L. Martínez, ".ªnalyzing the performance of classical consensus models in large scale group decision making: A comparative study"," *Applied Soft Computing*, vol. 67, pp. 677–690, 2018.

[6] I. Palomares, F. Estrella, L. Martinez, and F. Herrera, "Consensus under a fuzzy context: Taxonomy, analysis framework afryca and experimental case of study," *Information Fusion*, vol. 20, 11 2014.

[7] R. M. Rodríguez, Á. Labella, G. D. Tré, and L. Martínez, "A large scale consensus reaching process managing group hesitation," *Knowledge-Based Systems*, vol. 159, pp. 86–97, 2018.

[8] Á. Labella, H. Liu, R. M. Rodríguez, and L. Martínez, "A cost consensus metric for consensus reaching processes based on a comprehensive minimum cost model," *European Journal of Operational Research*, vol. 281, p. 316–331, 2020.

[9] H. Bustince, E. Barrenechea, and M. Pagola, "Relationship between restricted dissimilarity functions, restricted equivalence functions and normal en-functions: Image thresholding invariant," *Pattern Recognition Letters*, vol. 29, no. 4, pp. 525 – 536, 2008.

[10] sinbad2. Data set with 100 experts. [Online]. Available: https://sinbad2.ujaen.es/afryca/sites/default/files/dataset/EVA100Preferences.pdf

[11] L. Blouvshtein and D. Cohen-Or, "Outlier detection for robust multi-dimensional scaling," *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 9, pp. 2273–2279, 2018.

[12] A. Sîrbu, V. Loreto, V. Servedio, and F. TRIA, "Cohesion, consensus and extreme inormation in opinion dynamics," *Advances in Complex Systems*, vol. 16, no. 06, p. 1350035, 2013.

[13] A. Sîrbu, V. Loreto, and V. Servedio, "Opinion dynamics with disagreement and modulated information," *Journal of Statistical Physics*, vol. 151, pp. 218–237, 2013.

[14] R. R. Yager and F. E. Petry, "Hypermatching: Similarity matching with extreme values," *IEEE Transactions on Fuzzy Systems*, vol. 22, no. 4, pp. 949–957, 2014.

Estructuras de preferencia para relaciones recíprocas borrosas: caracterización y compatibilidad

Fabian Castiblanco
Faculty of Economics
Gran Colombia University
Bogotá, Colombia
fabianalberto.castiblanco@ugc.edu.co

Camilo Franco
Faculty of Engineering
Los Andes University
Bogotá, Colombia
c.franco31@uniandes.edu.co

J. Tinguaro Rodríguez
Faculty of Mathematics
Complutense University
Madrid, Spain
jtrodrig@mat.ucm.es

Javier Montero
Faculty of Mathematics
Complutense University
Madrid, Spain
monty@mat.ucm.es

***Abstract*—El empleo de relaciones borrosas $R : \mathbb{A}^2 \to [0,1]$ está ampliamente extendido de cara a representar grados de preferencia entre alternativas en $\mathbb{A}$. En el trabajo recientemente aceptado [1] se reflexiona en primer lugar sobre la aparente incompatibilidad entre las relaciones recíprocas y el marco estándar de preferencias borrosas establecido en [2], proponiendo a continuación un marco general para estudiar el significado preferencial de las relaciones recíprocas a partir de la semántica, bien establecida, de las relaciones y estructuras de preferencia del modelo estándar. Una consecuencia de este análisis es que es posible dotar a las relaciones recíprocas de una estructura de preferencia propia con dos relaciones, preferencia estricta y ausencia de preferencia. Además, es posible dotar de un significado unívoco a esta estructura sobre la base de las estructuras de preferencia estándar cuando se cumplen ciertas condiciones de compatibilidad entre ambos modelos.**

***Index Terms*—Relaciones de preferencia borrosa, relaciones recíprocas, estructuras de preferencia, semántica preferencial**

I. Introducción

Considérese un problema de decisión con conjunto de alternativas $\mathbb{A}$, y sea $R : \mathbb{A}^2 \to [0,1]$ una relación de preferencia borrosa sobre $\mathbb{A}$. R es una relación de preferencia *débil* si es reflexiva, esto es, si $R(a,a) = 1\ \forall a \in \mathbb{A}$, mientras que R es *estricta* si es antireflexiva, $R(a,a) = 0\ \forall a \in \mathbb{A}$, y R se dice *recíproca* cuando $R(a,b)+R(b,a) = 1\ \forall a,b \in \mathbb{A}$. Sean respectivamente $\mathcal{R}$ y $\mathcal{G}$ los conjuntos de todas las relaciones de preferencia borrosas débiles y recíprocas. Según se discute en [1], las relaciones de preferencia borrosas recíprocas presentan una incompatibilidad esencial con las relaciones de preferencia borrosa estándar, tanto débiles como estrictas, debido a que la condición de reciprocidad es incompatible con las propiedades de reflexividad y antireflexividad. Esto plantea dudas sobre el significado del predicado preferencial de una relación recíproca, así como sobre las situaciones preferenciales que un modelo de tipo recíproco puede expresar. Para estudiar esta cuestión, en [1] se plantea dotar a las relaciones recíprocas de una estructura de preferencia propia, y establecer el significado de esta en base al de las estructuras de preferencia estándar, siguiendo la formulación dada por Fodor y Roubens [2]. Esto se lleva a cabo en base a la posibilidad de obtener relaciones recíprocas a partir de relaciones débiles mediante transformaciones $F : \mathcal{R} \to \mathcal{G}$ representables en términos de una función *recíproca* $f : [0,1]^2 \to [0,1]$, tal que para $R \in \mathcal{R}$ y $a,b \in \mathbb{A}$ se cumple que $F(R)(a,b) = f(R(a,b), R(b,a)) = f(x,y)$. Esta herramienta permite probar que cualquier relación recíproca $G \in \mathcal{G}$ puede descomponerse como $G = P + 0.5(I+J)$, donde $\langle P,I,J\rangle$ denota cualquier estructura de preferencia de una relación débil $R \in \mathcal{R}$ tal que $R \in F_g^{-1}(G)$, donde la transformación $F_g : \mathcal{R} \to \mathcal{G}$ se representa mediante la función $g(x,y) = \frac{1+x-y}{2}$. Esta descomposición sugiere que cualquier relación recíproca puede expresarse en términos de una componente de preferencia estricta P y otra componente de ausencia de preferencia $L = I + J$, y que por tanto la estructura de preferencia de una relación recíproca $G \in \mathcal{G}$ debe estar compuesta por dos relaciones, P_G y L_G, respectivamente representando esas componentes de preferencia estricta y ausencia de preferencia. Sin embargo, esta descompoción no es única a menos que se cumpla una compatibilidad perfecta entre las estructuras de preferencia de las relaciones débiles y recíprocas implicadas. Las condiciones bajo las que se cumple esta compatibilidad son estudiadas en [1] mediante un enfoque axiomático, que conduce al planteamiento de un sistema de ecuaciones funcionales que solo puede verificarse mediante estructuras compatibles. Esto permite probar que esta compatibilidad solo es posible cuando la semántica de las relaciones débiles se modela mediante una preferencia estricta asimétrica. Además, se muestra también que el sistema de ecuaciones mencionado admite al menos dos soluciones, basadas en diferentes transformaciones entre relaciones débiles y recíprocas. Así, es posible concluir que las relaciones recíprocas pueden admitir diferentes semánticas compatibles con el marco estándar de preferencias borrosas, de modo parecido a cómo una relación de preferencia borrosa débil puede ser dotada de diferentes semánticas a partir de diferentes soluciones a las ecuaciones del modelo estándar.

Este trabajo ha sido parcialmente financiado por la Fundación Carolina (ayudas para estancias cortas postdoctorales), el Gobierno de España (proyecto PGC2018-096509-B-100) y la Universidad Complutense de Madrid (grupo de investigación 910149).

II. Definiciones y resultados

Recordemos en primer lugar que el modelo de preferencias borrosas estándar [2] se establece mediante un enfoque

axiomático que considera la existencia de funciones $p, i, j : [0,1]^2 \to [0,1]$, de manera que la estructura de preferencia $\langle P, I, J\rangle$ de una relación débil $R \in \mathcal{R}$ se obtiene como $P(a,b) = p(x,y)$, $I(a,b) = i(x,y)$ y $J(a,b) = j(a,b)$, donde $x = R(a,b)$ e $y = R(b,a)$, con $a, b \in \mathbb{A}$. Para representar la semántica mínima de preferencias, estas funciones p, i, j deben cumplir el sistema de ecuaciones funcionales

$$x = S_L(p(x,y), i(x,y)), \tag{1}$$

$$1 - y = S_L(p(x,y), j(x,y)). \tag{2}$$

donde S_L denota la t-conorma de Lukasiewicz bajo el automorfismo identidad. Existen múltiples soluciones al sistema anterior, siendo relevante para este trabajo la dada por

$$p(x,y) = \max\{x - y, 0\} \tag{3}$$

$$i(x,y) = \min\{x, y\} \tag{4}$$

$$j(x,y) = \min\{1 - x, 1 - y\} \tag{5}$$

que define una preferencia estricta asimétrica.

Asumiremos además que la representación de transformaciones $F : \mathcal{R} \to \mathcal{G}$ se lleva a cabo mediante funciones *recíprocas*, que se definen como funciones $f : [0,1]^2 \to [0,1]$ que verifican $f(x,y) + f(y,x) = 1$ para todo $x, y \in [0,1]$, y son continuas, monótonas no decrecientes en x (y por tanto no crecientes en y) y tales que $f(1,0) = 1$ y $f(0,1) = 0$. En [1] se prueban diversos resultados sobre estas funciones recíprocas.

Por otro lado, los principales resultados en [1] que sustentan la mencionada descomposición semántica general de cualquier relación recíproca $G \in \mathcal{G}$ son los siguientes:

Teorema 1. *Sea $\langle p, i, j\rangle$ una solución del sistema de ecuaciones del modelo estándar dado por Ecs.(1) y (2). Entonces, la función $f : [0,1]^2 \to [0,1]$ dada por $f(x,y) = p(x,y) + 0.5(i(x,y) + j(x,y))$ es una función recíproca, y se cumple que $f(x,y) = g(x,y) = \frac{1+x-y}{2}$.*

Corolario 1. *Sea $G \in \mathcal{G}$ una relación recíproca, y sea $R \in \mathcal{R}$ una relación débil tal que $G(a,b) = g(x,y)$ para todo $a, b \in \mathbb{A}$, donde $x = R(a,b)$ e $y = R(b,a)$. Si $\langle P, I, J\rangle$ es una estructura de preferencia asociada a R mediante alguna solución del sistema dado por las Ecs.(1) y (2), entonces $G(a,b) = P(a,b) + 0.5(I(a,b) + J(a,b))$ para todo $a, b \in \mathbb{A}$.*

Así pues, toda relación recíproca puede ser descompuesta en términos de dos componentes, preferencia estricta P y ausencia de preferencia (estricta) $L = I + J$, en tanto I y J recogen las situaciones preferenciales que no pueden ser asociadas a preferencia entre alternativas. Esto lleva a asociar una nueva relación borrosa L a esa componente de ausencia de preferencia, dada por $L(a,b) = S_L(I(a,b), J(a,b))$, para todo $a, b \in \mathbb{A}$. Es fácil comprobar que esta relación L es reflexiva y simétrica. Además, es posible reestablecer la descomposición anterior en términos de disyunción entre preferencia estricta P y ausencia de preferencia L:

Corolario 2. *En las condiciones del Corolario 1, se cumple que $G(a,b) = S_L(P(a,b), 0.5L(a,b))$.*

Estos resultados sugieren que es natural asociar a una relación recíproca $G \in \mathcal{G}$ una estructura de preferencia $\langle P_G, L_G\rangle$. No obstante, en principio la descomposición de G en las relaciones P_G y L_G no es única, en tanto depende de la descomposición de una relación débil $R \in \mathcal{R}$ tal que $G = F_g(R)$ en la estructura $\langle P, I, J\rangle$, que es posible realizar mediante cualquier solución de las Eqs.(1) y (2). Esto además conlleva que la estructura $\langle P_G, L_G\rangle$ no sea necesariamente obtenible a partir de G. Para estudiar la posibilidad de que la estructura $\langle P_G, L_G\rangle$ pueda ser obtenida directamente a partir de G, y que al mismo tiempo sea armónica con la semántica de la estructura de preferencia débil $\langle P, I, J\rangle$ que la dota de significado, en [1] se recurre a un enfoque axiomático que asume la existencia de funciones $p_G, l_G : [0,1] \to [0,1]$ tales que $P_G(a,b) = p_g(u)$ y $L_G(a,b) = l_G(u)$ para todo $a, b \in \mathbb{A}$ y $u = G(a,b)$, con p_G no decreciente y l_G simétrica respecto a 0.5, esto es, $l_G(u) = l_G(1-u)$, y que conduce a plantear el siguiente sistema de ecuaciones funcionales:

$$S_L(p_R(x,y), i_R(x,y)) = x, \tag{6}$$

$$S_L(p_R(x,y), j_R(x,y)) = 1 - y, \tag{7}$$

$$u = f(x,y), \tag{8}$$

$$p_G(u) = p_R(x,y), \tag{9}$$

$$l_G(u) = S_L(i_R(x,y), j_R(x,y)), \tag{10}$$

$$u = A(p_G(u), l_G(u)), \tag{11}$$

para todo $x, y \in [0,1]$, y donde $A : \{(p_G(u), l_G(u)) | u \in [0,1]\} \to [0,1]$ es una función continua que permite recuperar la relación recíproca G a partir de su estructura de preferencia. El principal resultado en [1] sobre este sistema es el siguiente:

Teorema 2. *El sistema dado por Ecs.(6)-(11) admite solución si y solo si p_R, i_R, j_R vienen dadas por Ecs.(3)-(5).*

Una solución particular del sistema anterior, cuando la función recíproca empleada es $f = g$, viene dada como:

Teorema 3. *Si una solución del sistema Ecs.(6)-(11) es tal que p_R, i_R, j_R vienen dadas por Ecs.(3)-(5) y $f = g$, entonces $p_G(u) = \max\{2u - 1, 0\}$, $l_G(u) = 1 - |2u - 1|$ y $A(p_G(u), l_G(u)) = S_L(p_G(u), 0.5 l_G(u))$ para todo $u \in [0,1]$.*

En [1] se establecen otros resultados adicionales, como que el sistema anterior admite solución solo si f es estrictamente creciente en x y es invariante en la dirección (1,1) de $[0,1]^2$, o que la función A es la dada en el teorema anterior si y solo si $f = g$, y se prueba la existencia de al menos otra solución al sistema referido.

References

[1] F. Castiblanco, C. Franco, J. T. Rodríguez and J. Montero (2021) "A characterization of reciprocal fuzzy preference structures and its compatibility with standard fuzzy preference structures," Fuzzy Sets and Systems, IN PRESS https://doi.org/10.1016/j.fss.2021.02.001.

[2] J. Fodor, M. Roubens (1994) Fuzzy Preference Modelling and Multicriteria Decision Support. Dordrecht: Kluwer Academic Publishers.

Generalización de intervalos en matrices de riesgo imprecisas

María Martínez
Dept. of Statistics and O.R.
University of Oviedo
Oviedo, Spain
UO257879@uniovi.es

Juan Baz
Dept. of Statistics and O.R.
University of Oviedo
Gijón, Spain
bazjuan@uniovi.es

Raul Pérez-Fernández
Dept. of Statistics and O.R.
University of Oviedo
Gijón, Spain
perezfernandez@uniovi.es

Susana Montes
Dept. of Statistics and O.R.
University of Oviedo
Oviedo, Spain
montes@uniovi.es

***Resumen*—A pesar de que la fiabilidad humana puede tener mucha imprecisión asociada, debido a la incertidumbre natural relacionada con las decisiones tomadas por las personas, una de las principales herramientas en esta área, la matriz de riesgo, se define de manera muy precisa. Esto hace que los expertos no puedan tener ninguna duda a la hora de clasificar un posible error humano. Para evitar este problema, aquí se propone una relajación de la matriz de riesgo, en la que la valoración de los riesgos se dará no solo mediante números, sino también utilizando intervalos o incluso una generalización de estos, que llamaremos cajas. Dicho concepto es analizado en detalle, así como los órdenes obtenidos para el conjunto de cajas.**

***Index Terms*—matriz de riesgo, órdenes entre intervalos, caja, órdenes entre cajas, métodos de elección**

I. Introducción

Las técnicas de análisis de fiabilidad se enfocaron inicialmente en aspectos técnicos del diseño y calidad de la maquinaria. Sin embargo, algunas investigaciones demostraron que el error humano era la causa más común de fallo en muchas situaciones (ver, por ejemplo, [1], [4], [7]). Desde que IEEE publicó un informe sobre fiabilidad humana en 1972, han surgido muchos trabajos en esta temática (ver, por ejemplo, [2], [8], [12], [14], [16], [26]).

Una herramienta fundamental es la matriz de evaluación de riesgos (ver [9]). Esta herramienta facilita la clasificación de diferentes tipos de errores en función de su riesgo asociado. Esta clasificación puede ayudar a priorizar los diferentes tipos de error.

Existen distintas versiones de matrices de evaluación de riesgos, pero en todas ellas los decisores se ven obligados a elegir entre los diferentes niveles de consecuencia/frecuencia, a pesar de que están tomando una decisión subjetiva y, en muchos casos, la elección del nivel apropiado se convierte en una tarea difícil. Para considerar un procedimiento más flexible, proponemos una metodología en la que los decisores pueden expresar valores intermedios. Bajo esta consideración, el valor asociado con un tipo de error ya no es un número sino un intervalo de la línea real o incluso una generalización de este que llamaremos caja. Así, para medir el riesgo asociado a los diferentes tipos de error, es necesario establecer un orden entre cajas, que generalizará los órdenes considerados entre números e intervalos.

Así, la estructura de este trabajo es la siguiente: la sección 2 ofrece una visión general de los conceptos básicos necesarios para la comprensión del mismo. En la sección 3 introduciremos el concepto de caja, así como distintas formas de ordenarlas, lo que nos permitirá acabar proponiendo una matriz de riesgo imprecisa, que era el objetivo fundamental del trabajo. Para ver la forma de utilizar la metodología propuesta, planteamos un ejemplo de aplicación en la sección 4, donde la información proporcionada por los distintos órdenes es fusionada mediante métodos de ranking. Finalizamos con algunas conclusiones y puntos abiertos.

II. Conceptos básicos

En esta sección, se recuerdan algunos conceptos básicos sobre matrices de riesgo y órdenes entre intervalos. Sirve además para fijar las notaciones que se van a utilizar a lo largo del trabajo.

A. Matriz de evaluación de riesgos

La matriz de riesgo es un elemento que permite cuantificar los riesgos disminuyendo el nivel de subjetividad en el momento de su evaluación, siempre que la parametrización y asignación de valores a los indicadores esté debidamente fundamentada. Son un instrumento que nace con la necesidad de poder cuantificar los errores y saber así sobre cuáles deberemos actuar con mayor rapidez.

Para obtener los valores de dicha matriz se deberá estudiar cuál es la frecuencia y las consecuencias de cada uno de los errores a examinar. Veamos cuál es su definición formal.

Definición 1: [9], [19] Sea $\mathcal{M}$ una matriz de dimensiones $n \times m$, se dice **matriz de riesgo** a la combinación de ciertas consecuencias (asociadas a las columnas de la matriz), ocurriendo en un cierto escenario y con una cierta probabilidad (asociadas a las filas de la matriz), lo cual significa que se necesitan solamente dos entradas para construir la matriz de riesgo.

- Los valores de dicha matriz reciben el nombre de **valores de riesgo** y quedarán determinados por los ejes de la matriz de riesgo, es decir por las consecuencias y la probabilidad de las mismas.

Este estudio ha sido parcialmente patrocinado por el programa español MINECO (TIN-2017-87600-P: J. Baz y R. Pérez-Fernández; PGC2018-098623-B-I00: S. Montes).

- Las consecuencias, la probabilidad y los valores de riesgo pueden estar divididos en diferentes niveles, según el caso particular con el que estudiemos dicha matriz.
- El proceso del cálculo de los valores de riesgo viene presentado por la implicación lógica: si la probabilidad es p y consecuencia es c, entonces el riesgo es r.

A pesar de esta definición común, la estructura de la matriz de riesgo es muy variable, pudiendo variar tanto sus dimensiones, como la posible nomenclatura. En el caso particular de matriz de riesgo aquí considerada (ver [18]), el eje de la frecuencia lo podremos categorizar como improbable (I), poco probable (Pp), posible (Pos), probable (P) y muy probable (Mp) y el eje de las consecuencias lo podremos clasificar como poco (P), normal (N), grande (G), elevada (E) y extrema (Ex). Las filas *Frec.* indican la frecuencia o probabilidad de ocurrencia y las columnas *Consec.* indican las consecuencias. Así pues, la matriz de riesgo estándar que vamos a considerar es la mostrada en la tabla I.

TABLE I
MATRIZ DE RIESGO ESTÁNDAR.

Frec. \ Cons.	Poco	Normal	Grande	Elevada	Extrema
Muy probable	5	10	15	20	25
Probable	4	8	12	16	20
Posible	3	6	9	12	15
Poco probable	2	4	6	8	10
Improbable	1	2	3	4	5

Típicamente se asigna un código de colores a los valores de la matriz de riesgos, según cual sea su valor. En concreto, se considera el color verde para valores de riesgo menores o igual que 3, el amarillo para valores mayores que 3 y menores o iguales que 7, el color naranja para valores mayores que 7 pero no mayores que 12 y el rojo para el resto. Es evidente que la zona con mayor riesgo se representa de color rojo. En esta zona están los problemas graves, aquellos sobre los que debemos encontrar una rápida solución. La zona naranja muestra los problemas que tienen una importancia elevada. La zona media, de color amarilla, es aquella donde se encuentran los problemas que tienen menos relevancia pero aún así son importantes. La zona de bajo riesgo, en color verde, contiene los riesgos controlados. Es habitual que no se actúe sobre los problemas que están en esta área.

B. *Ordenación de intervalos en* $\mathbb{R}$

En la subsección anterior hemos visto que los errores se cuantifican mediante números reales entre 1 y 25. No obstante, a la hora de flexibilizar esta matriz para recoger mejor la incertidumbre asociada a la toma de decisiones, nos van a aparecer en la misma intervalos. Recordemos que dados x_1, $x_2 \in \mathbb{R}$ con $x_1 \leq x_2$, llamaremos intervalo (cerrado) al conjunto definido como $X = [x_1, x_2] = \{y \in \mathbb{R} \ / \ x_1 \leq y \leq x_2\}$, denotando por $I(\mathbb{R})$ el conjunto de todos los intervalos cerrados en el conjunto de los números reales, es decir, $I(\mathbb{R}) = \{[x_1, x_2] \ / \ x_1, x_2 \in \mathbb{R} \text{ con } x_1 \leq x_2\}$.

Es sabido que si bien los números reales forman un conjunto totalmente ordenado con el orden usual ($a < b$ si $b - a$ es un número positivo), encontrar un orden total en el conjunto de los intervalos no es un problema tan inmediato. Así, en la literatura se han introducido diversas formas de comparar intervalos. En concreto, algunos ejemplos son los siguientes: orden débil [5], Maximin [22], [24], Maximax [21], Dominancia de intervalos [13], Hurwicz [17] y el orden producto (también llamado *lattice order* en inglés) [15]. De todos ellos, solo el orden producto es un verdadero orden. Recordemos que viene definido por: $X \leq_{LO} Y \Leftrightarrow x_1 \leq y_1$ y $x_2 \leq y_2$. No obstante, y a pesar de ser un orden muy natural y habitual entre intervalos, es evidente que considerando dicho orden hay intervalos incomparables. Para evitar este problema, buscamos órdenes que sean completos y refinen el orden producto. Esto es posible gracias al concepto de orden admisible (orden total que refina el orden producto) introducido por Bustince et al. (ver [6]).

Dados X, $Y \in I(\mathbb{R})$ con $X = [x_1, x_2]$ e $Y = [y_1, y_2]$, son órdenes admisibles sobre $I(\mathbb{R})$: 1) Orden lexicográfico 1 : $X \leq_{Lex1} Y$ si y solo si $x_1 < y_1$ o $(x_1 = y_1$ y $x_2 \leq y_2)$; 2) Orden lexicográfico 2: $X \leq_{Lex2} Y$ si y solo si $x_2 < y_2$ o $(x_2 = y_2$ y $x_1 \leq y_1)$ y 3) Orden de Xu y Yager ([25]): $X \leq_{XY} Y$ si y solo si $x_1 + x_2 < y_1 + y_2$ o $(x_1 + x_2 = y_1 + y_2$ y $x_2 - x_1 \leq y_2 - y_1)$.

III. MATRIZ DE RIESGO IMPRECISA

En una matriz de riesgo clásica, el decisor tiene que decidir por una frecuencia poco probable o posible y no se permite una opción intermedia. Sin embargo, podría ser muy natural una clasificación del riesgo entre poco probable y posible. En ese caso, podría ser lógico considerar que el valor de riesgo es un número entre 8 y 12 si la consecuencia es elevada. Por lo tanto, el valor de riesgo podría venir dado por el intervalo cerrado $[8, 12]$.

Puede ocurrir una situación similar con respecto a la consecuencia. Por tanto, podríamos considerar situaciones intermedias y llegamos a una propuesta de matriz de riesgo 9×9, que permite opciones intermedias para la frecuencia y la consecuencia, tal como la representada en la tabla II.

TABLE II
PRIMERA APROXIMACIÓN A LA MATRIZ DE RIESGO IMPRECISA.

Frec. \ Consec.	P	PN	N	NG	G	GE	E	EEx	Ex
MP	5	[5,10]	10	[10,15]	15	[15,20]	20	[20,25]	25
MPP	[4,5]		[8,10]		[12,15]		[16,20]		[20,25]
P	4	[4,8]	8	[8,12]	12	[12,16]	16	[16,20]	20
PPos	[3,4]		[6,8]		[9,12]		[12,16]		[15,20]
Pos	3	[3,6]	6	[6,9]	9	[9,12]	12	[12,15]	15
PosPP	[2,3]		[4,6]		[6,9]		[8,12]		[10,15]
PP	2	[2,4]	4	[4,6]	6	[6,8]	8	[8,10]	10
PPI	[1,2]		[2,4]		[3,6]		[4,8]		[5,10]
I	1	[1,2]	2	[2,3]	3	[3,4]	4	[4,5]	5

En la tabla II, se ha denotado con las letras correspondientes a las dos clases que la delimitan a cualquier categoría intermedia. Por ejemplo, si la frecuencia es entre poco probable y posible, se representa por PosPp, si la consecuencia es entre elevada y extrema, se denota por EEx, etc.

En el caso de que exista imprecisión en la consecuencia, pero la frecuencia sea fija, ya tendríamos resuelto el problema

de clasificar un riesgo, tal como hemos visto en la tabla II. En este caso el valor de riesgo sería un intervalo. Lo mismo ocurre si la consecuencia es fija y la imprecisión está en la frecuencia. Puesto que todo número x se puede identificar con el intervalo $[x, x]$, podríamos considerar los órdenes entre intervalos vistos en la sección anterior para ordenar los distintos errores según sus valores de riesgo. El problema viene cuando tanto la valoración de las consecuencias, como la de las frecuencias, es imprecisa. En tal caso no hemos podido valorar el riesgo, tal como se ha visto en la tabla II. Vamos a centrarnos en una zona determinada de la misma, para ir analizando la solución dada para este problema. En concreto nos centramos en la zona correspondiente al caso de consecuencias entre elevadas y extremas y de frecuencias entre poco probable y posible, la cual aparece ampliada en la tabla III.

TABLE III
ZOOM DE LA MATRIZ DE RIESGO.

		Consecuencia		
		E	EEx	Ex
Frecuencia	Pos	12	$[12, 15]$	15
	PosPp	$[8, 12]$	?	$[10, 15]$
	Pp	8	$[8, 10]$	10

Para la celda restante, podríamos pensar en considerar nuevamente un intervalo. Sin embargo, lo que tiene que ocurrir es que el valor debe estar entre $[8, 10]$ y $[12, 15]$ y además entre $[8, 12]$ y $[10, 15]$. Para recoger esta idea, aparece un nuevo concepto como intersección de dos "intervalos" de intervalos.

Definición 2: Sean a, b, c, d cuatro números reales con $a \leq b \leq c \leq d$. El conjunto $\{[x, y] \in I(\mathbb{R}) / [a, c] \leq_{LO} [x, y] \leq_{LO} [b, d], [a, b] \leq_{LO} [x, y] \leq_{LO} [c, d]\}$ se llama caja con extremos a, b, c y d y se denota por $[a, b, c, d]$.

Por lo tanto, está claro que cualquier elemento en la caja $[a, b, c, d]$ es un intervalo tal que está entre el intervalo $[a, b]$ y el intervalo $[c, d]$ y, al mismo tiempo, está entre el intervalo $[a, c]$ y el intervalo $[b, d]$ con respecto al orden producto. Denotaremos por $\mathcal{C}(\mathbb{R})$ al conjunto de todas las cajas en $\mathbb{R}$, es decir, $\mathcal{C}(\mathbb{R}) = \{[a, b, c, d] / a, b, c, d \in \mathbb{R}, a \leq b \leq c \leq d\}$. A continuación, presentamos una definición equivalente del concepto de caja.

Proposición 1: Sea $[a, b, c, d]$ en $\mathcal{C}(\mathbb{R})$. Se tiene que:

$$[a, b, c, d] = \{[x, y] \in I(\mathbb{R}) / a \leq x \leq b, c \leq y \leq d\}$$

De esta definición equivalente se puede deducir que una caja es un tipo particular de hiperrectángulo bidimensional o 2-ortótropo (ver [10]). Por otro lado, considerando ahora la identificación entre la caja $[a, b, c, d]$ y la 4-tupla (a, b, c, d), el concepto de caja coincide con el de intervalo 4-dimensional, tal como ha sido definido por Bedregal et al. en [3]. En dicho trabajo se pone además de manifiesto que el conjunto de los intervalos 2-dimensionales coincide con el conjunto de los intervalos cerrados y el de los intervalos 1-dimensionales con el de los números reales. Así pues, todos los valores que tenemos hasta el momento en la matriz de riesgo (ver tabla II) puede considerarse que son cajas. Además, podríamos considerar cajas adecuadas para llenar los espacios vacíos en esta matriz. Por ejemplo, el valor restante en el ejemplo considerado en la tabla III sería $[8, 10, 12, 15]$. Podríamos repetir este procedimiento a lo largo de todas las celdas vacías de la matriz de la tabla II. Sin embargo, para poder colorear esta matriz, debemos definir un orden apropiado en $\mathcal{C}(\mathbb{R})$. Una primera propuesta podría ser la siguiente.

Definición 3: Sea $[a, b, c, d]$ y $[\bar{a}, \bar{b}, \bar{c}, \bar{d}]$ en $\mathcal{C}(\mathbb{R})$. Diremos que $[a, b, c, d]$ es menor o igual que $[\bar{a}, \bar{b}, \bar{c}, \bar{d}]$ con respecto al orden producto, y se denota por $[a, b, c, d] \leq_{LO} [\bar{a}, \bar{b}, \bar{c}, \bar{d}]$, si y solo si $a \leq \bar{a}$, $b \leq \bar{b}$, $c \leq \bar{c}$ y $d \leq \bar{d}$.

Es fácil probar que $\leq_{LO}$ es un orden en $\mathcal{C}(\mathbb{R})$, pero no es una relación completa. Por ejemplo, $[2, 3, 4, 5] \not\leq_{LO} [2, 2, 5, 5]$ y $[2, 2, 5, 5] \not\leq_{LO} [2, 3, 4, 5]$. Como necesitamos ordenar todos los elementos en la matriz de riesgo, necesitamos definir un orden total en $\mathcal{C}(\mathbb{R})$. A partir de las ideas de los órdenes lexicográficos en $L(\mathbb{R})$ podemos obtener 24 órdenes distintos en $\mathcal{C}(\mathbb{R})$. Comenzamos con el de tipo 1, que nos servirá para generar el resto.

Definición 4: Sea $[a, b, c, d]$ y $[\bar{a}, \bar{b}, \bar{c}, \bar{d}]$ en $\mathcal{C}(\mathbb{R})$. Diremos que $[a, b, c, d]$ es menor o igual que $[\bar{a}, \bar{b}, \bar{c}, \bar{d}]$ con respecto al orden lexicográfico tipo 1, que se denota por $[a, b, c, d] \leq_{1234} [\bar{a}, \bar{b}, \bar{c}, \bar{d}]$, si y solo si se cumple una de las condiciones siguientes: $a < \bar{a}$ o $(a = \bar{a}, b < \bar{b})$ o $(a = \bar{a}, b = \bar{b}, c < \bar{c})$ o $(a = \bar{a}, b = \bar{b}, c = \bar{c}, d \leq \bar{d})$.

Se puede probar que la relación en $\mathcal{C}(\mathbb{R})$ es un orden total que refina el orden producto.

Puesto que en dicha definición no hemos considerado las relaciones entre los cuatro elementos que describen una caja y cualquier caja $[a, b, c, d]$ se puede identificar con una 4-tupla (a, b, c, d) en $\mathbb{R}^4$, el orden lexicográfico 1 en $\mathcal{C}(\mathbb{R})$ se podría generalizar de forma inmediata a $\mathbb{R}^4$ y también en ese espacio sería un orden lineal. De hecho, vamos a utilizarlo como generador de órdenes lineales en ese espacio, tal como puede verse en el siguiente resultado.

Proposición 2: Sea $\mathcal{A}$ una matriz cualquiera de rango completo con $\mathcal{A} \in \mathcal{M}_{4\times4}$. Se tiene que la relación binaria $\mathcal{R}$ definida sobre $\mathbb{R}^4 \times \mathbb{R}^4$ como $(a, b, c, d)\mathcal{R}(\bar{a}, \bar{b}, \bar{c}, \bar{d})$ si y solo si $\mathcal{A}(a, b, c, d)' \leq_{1234} \mathcal{A}(\bar{a}, \bar{b}, \bar{c}, \bar{d})'$ para cualesquiera $(a, b, c, d), (\bar{a}, \bar{b}, \bar{c}, \bar{d}) \in \mathbb{R}^4$, es un orden total en $\mathbb{R}^4$.

Este resultado puede considerarse una generalización inmediata del demostrado para el intervalo $[0, 1]$ en [11]. Además es inmediato que cualquier orden generado de esta forma, puede ser restringido al conjunto de cajas $\mathcal{C}(\mathbb{R})$. En concreto, es evidente que según vayamos considerando las 24 matrices de rango completo con un 1 en cada fila y el resto de los elementos nulos, podríamos generar 24 órdenes lineales distintos sobre $\mathcal{C}(\mathbb{R})$.

Corolario 1: Sea $\{i, j, k, l\}$ una permutación cualquiera del conjunto $\{1, 2, 3, 4\}$ y sea $\mathcal{A}_{ijkl}$ una matriz en $\mathcal{M}_{4\times4}$ tal que todos sus elementos son nulos salvo $a_{1i} = a_{2j} = a_{3k} = a_{4l} = 1$. La relación $\leq_{ijkl}$ definida como $[a, b, c, d] \leq_{ijkl} [\bar{a}, \bar{b}, \bar{c}, \bar{d}]$ si y solo si $\mathcal{A}_{ijkl}(a, b, c, d)' \leq_{1234} \mathcal{A}_{ijkl}(\bar{a}, \bar{b}, \bar{c}, \bar{d})')$, para cualesquiera $[a, b, c, d], [\bar{a}, \bar{b}, \bar{c}, \bar{d}] \in \mathcal{C}(\mathbb{R})$, es un orden total en $\mathcal{C}(\mathbb{R})$.

Si aplicamos la proposición 2 para la matriz

$$\mathcal{A} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ -1 & 1 & -1 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

obtenemos la generalización del orden de Xu-Yager para cajas.

Proposición 3: Dadas $[a, b, c, d]$ y $[\bar{a}, \bar{b}, \bar{c}, \bar{d}]$ dos cajas en $\mathcal{C}(\mathbb{R})$. La relación $\leq_{XY}$ sobre $\mathcal{C}(\mathbb{R})$ definida como: $[a, b, c, d] \leq_{XY} [\bar{a}, \bar{b}, \bar{c}, \bar{d}]$ si y sólo si verifica alguna de las siguientes condiciones:

- $a + b + c + d < \bar{a} + \bar{b} + \bar{c} + \bar{d}$
- $a + b + c + d = \bar{a} + \bar{b} + \bar{c} + \bar{d}$ y $(b - a) + (d - c) < (\bar{b} - \bar{a}) + (\bar{d} - \bar{c})$
- $a + b + c + d = \bar{a} + \bar{b} + \bar{c} + \bar{d}$, $(b - a) + (d - c) = (\bar{b} - \bar{a}) + (\bar{d} - \bar{c})$ y $a < \bar{a}$
- $a + b + c + d = \bar{a} + \bar{b} + \bar{c} + \bar{d}$, $(b - a) + (d - c) = (\bar{b} - \bar{a}) + (\bar{d} - \bar{c})$, $a = \bar{a}$ y $c \leq \bar{c}$

es un orden total que refina el orden producto.

También se puede probar que si nos restringimos a intervalos estos órdenes corresponden con sus homónimos en $L(\mathbb{R})$ y si nos restringimos a números, con el orden habitual sobre la recta real. Evidentemente una caja que es menor que otra con respecto a un orden, puede no serlo con respecto al otro. Si consideramos el orden lexicográfico tipo 1 y los colores verde ($\leq_{1234}$ 3), amarillo ($>_{1234}$ 3 y $\leq_{1234}$ 7), naranja ($>_{1234}$ 7 y $\leq_{1234}$ 12) y rojo ($>_{1234}$ 12), obtenemos la matriz de riesgo imprecisa que puede verse en la tabla IV.

TABLE IV
MATRIZ DE RIESGO IMPRECISA.

Frec. \ Cons.	P	PN	N	NG	G	GE	E	EEx	Ex
Mp	5	[5,10]	10	[10,15]	15	[15,20]	20	[20,25]	25
MpP	[4,5]	[4,5,8,10]	[8,10]	[8,10,12,15]	[12,15]	[12,15,16,20]	[16,20]	[16,20,20,25]	[20,25]
P	4	[4,8]	8	[8,12]	12	[12,16]	16	[16,20]	20
PPos	[3,4]	[3,4,6,8]	[6,8]	[6,8,9,12]	[9,12]	[9,12,12,16]	[12,16]	[12,15,16,20]	[15,20]
Pos	3	[3,6]	6	[6,9]	9	[9,12]	12	[12,15]	15
PosPp	[2,3]	[2,3,4,6]	[4,6]	[4,6,6,9]	[6,9]	[6,8,9,12]	[8,12]	[8,10,12,15]	[10,15]
Pp	2	[2,4]	4	[4,6]	6	[6,8]	8	[8,10]	10
PpI	[1,2]	[1,2,2,4]	[2,4]	[2,3,4,6]	[3,6]	[3,4,6,8]	[4,8]	[4,5,8,10]	[5,10]
I	1	[1,2]	2	[2,3]	3	[3,4]	4	[4,5]	5

Si prefiriésemos trabajar con todos los órdenes a la vez, combinándolos, podríamos utilizar métodos de elección como el de Borda o Concorcet. Vamos a ver el procedimiento completo mediante un ejemplo.

IV. CASO DE ESTUDIO

Veremos de qué manera podemos aplicar estas formas de ordenar los distintos tipos de errores en un ejemplo ilustrativo (ver [23]). El objetivo es mostrar de qué forma quedan clasificados los errores según el orden que utilicemos. Así, podremos aplicar toda la teoría que hemos desarrollado para poder ver cuál es su utilidad real. En particular, estudiaremos y clasificaremos los errores más comunes que puede cometer un conductor de autobús. Podemos considerar los siguientes eventos:

- E1. Cruzar un semáforo en ámbar.
- E2. Cruzar un semáforo en rojo.
- E3. Conducir bajo los efectos de alcohol o de drogas.
- E4. No poner el intermitente al girar.
- E5. Saludar a otros conductores.
- E6. Confundirse en el cambio más de 50 céntimos.
- E7. Confundirse en el cambio menos de 50 céntimos.

Ahora, deberemos ver cuál es la frecuencia con la que se produce cada uno de los eventos, así como ver cuáles son las consecuencias de los mismos. Las frecuencias y las consecuencias de cada error se pueden determinar, por ejemplo, a partir de estadísticas que relacionen la asiduidad y el impacto que han tenido a lo largo de cierto periodo de tiempo. En base a esos datos, se podrían clasificar en las distintas categorías de las frecuencias y consecuencias (con la posibilidad de considerar intervalos).

En la tabla V aparece resumida toda la información: la etiqueta que relaciona cada error, con unas frecuencias y consecuencias que se han considerado razonables.

TABLE V
ESTIMACIÓN DE LAS FRECUENCIAS Y DE LAS CONSECUENCIAS DE CADA EVENTO.

Evento	Frecuencias	Consecuencias
E1	MpP	GE
E2	Pos	Ex
E3	PpI	Ex
E4	PosPp	EEx
E5	Mp	P
E6	PpI	PN
E7	PosPp	P

La notación utilizada es la misma que se ha utilizado en la matriz de riesgo imprecisa de la tabla IV.

Una vez obtenida la información de la tabla V, debemos considerar dicha matriz de riesgo para asignar a cada uno de los eventos (errores a cuantificar) el valor de riesgo asociado a los mismos, que vendrá dado mediante una caja, recordando que los números y los intervalos pueden ser vistos como tipos particulares de cajas. Así, dichas valoraciones son:

- E1. Cruzar un semáforo en ámbar: $\mathcal{C}_{E1} = [12, 15, 16, 20]$.
- E2. Cruzar un semáforo en rojo: $\mathcal{C}_{E2} = 15$.
- E3. Conducir bajo los efectos de alcohol o de drogas: $\mathcal{C}_{E3} = [5, 10]$.
- E4. No poner el intermitente al girar: $\mathcal{C}_{E4} = [8, 10, 12, 15]$.
- E5. Saludar a otros conductores: $\mathcal{C}_{E5} = 5$.
- E6. Confundirse en el cambio más de 50 céntimos: $\mathcal{C}_{E6} = [1, 2, 2, 4]$.
- E7. Confundirse en el cambio menos de 50 céntimos: $\mathcal{C}_{E7} = [2, 3]$.

Si aplicamos cada uno de los 24 órdenes lexicográficos que hemos introducido, la ordenación entre los distintos errores se repite en muchos de ellos, tal como puede verse en la tabla VI.

Nótese que tenemos solamente tres ordenaciones posibles. Aún así, se observa como la elección del orden entre cajas hace que la clasificación de los errores varíe de un caso a otro. Si queremos fusionar la información de varios órdenes, para no dar más relevancia a unas componentes de la caja que a otras, podríamos fusionar las distintas ordenaciones mediante métodos de elección. En particular, estudiaremos los métodos de Borda y de Condorcet. El primero se basa en la posición y el

TABLE VI
ORDENACIONES OBTENIDAS SEGÚN LOS DISTINTOS ÓRDENES LEXICOGRÁFICOS.

Frec.	Ranking
8	$\mathcal{C}_{E1} \geq \mathcal{C}_{E2} \geq \mathcal{C}_{E4} \geq \mathcal{C}_{E3} \geq \mathcal{C}_{E5} \geq \mathcal{C}_{E7} \geq \mathcal{C}_{E6}$
8	$\mathcal{C}_{E1} \geq \mathcal{C}_{E2} \geq \mathcal{C}_{E4} \geq \mathcal{C}_{E3} \geq \mathcal{C}_{E5} \geq \mathcal{C}_{E6} \geq \mathcal{C}_{E7}$
8	$\mathcal{C}_{E2} \geq \mathcal{C}_{E1} \geq \mathcal{C}_{E4} \geq \mathcal{C}_{E3} \geq \mathcal{C}_{E5} \geq \mathcal{C}_{E7} \geq \mathcal{C}_{E6}$

segundo en la comparación por pares. En la mayoría de casos ambos criterios actúan de igual forma, aunque puede haber casos en los que la ordenación sea completamente contraria. Veremos a continuación qué es lo que sucede en este ejemplo.

A. *Ranking Borda Count*

Existen distintos rankings para clasificar órdenes, en este caso nos centraremos en el Ranking de Borda Count.

Para hacer una clasificación de todos estos órdenes que hemos señalado previamente, usaremos dicho ranking para fusionar la información dada por todos los órdenes considerados.

En este método (ver [20] para más información al respecto), basta mirar cuál es la frecuencia de cada una de las ordenaciones posibles. La representación usual es la que aparece en la matriz $\mathcal{O}$, llamada **matriz de votación**, en la que el elemento o_{ij} representa el número de veces que $a_i \geq a_j$. A partir de la matriz $\mathcal{O}$, bastará sumar por filas obteniendo un valor α_i. Para obtener el ranking final basta hacer una clasificación natural con los valores α_i obtenidos.

La matriz de votación asociada a este ejemplo puede verse en la tabla VII.

TABLE VII
MATRIZ DE VOTACIÓN.

$\mathcal{O}$	$\mathcal{C}_{E1}$	$\mathcal{C}_{E2}$	$\mathcal{C}_{E3}$	$\mathcal{C}_{E4}$	$\mathcal{C}_{E5}$	$\mathcal{C}_{E6}$	$\mathcal{C}_{E7}$
$\mathcal{C}_{E1}$	0	16	24	24	24	24	24
$\mathcal{C}_{E2}$	8	0	24	24	24	24	24
$\mathcal{C}_{E3}$	0	0	0	0	24	24	24
$\mathcal{C}_{E4}$	0	0	24	0	24	24	24
$\mathcal{C}_{E5}$	0	0	0	0	0	24	24
$\mathcal{C}_{E6}$	0	0	0	0	0	0	8
$\mathcal{C}_{E7}$	0	0	0	0	0	16	0

La puntuación que nos ofrece el ranking de Borda se denota por

$$\alpha_{E_i} = \sum_{j=1}^{n} o_{ij}$$

donde el elemento o_{ij} representa la i-ésima fila y la j-ésima columna de la matriz $\mathcal{O}$.

Así, las puntuaciones de Borda serían las siguientes: $\alpha_{E_1} = 136$ y $\alpha_{E_2} = 128$, $\alpha_{E_3} = 72$, $\alpha_{E_4} = 96$, $\alpha_{E_5} = 48$, $\alpha_{E_6} = 8$, $\alpha_{E_7} = 16$. Se obtendría así el ranking según el Ranking de Borda Count por medio de la ordenación de estos números de menor a mayor:

$$\mathcal{C}_{E6} \leq \mathcal{C}_{E7} \leq \mathcal{C}_{E5} \leq \mathcal{C}_{E3} \leq \mathcal{C}_{E4} \leq \mathcal{C}_{E2} \leq \mathcal{C}_{E1}$$

Vemos en la figura 1 cómo queda la representación gráfica si usamos un diagrama de Hasse.

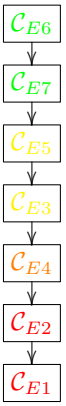


Fig. 1. Órdenes lexicográficos (ordenación por Borda).

Observamos como el diagrama tiene un aspecto lineal. El evento que tiene una mayor importancia y que domina a todos los demás es el evento que queda en la parte inferior, en este caso es el evento $\mathcal{C}_{E1}$, como ya hemos visto.

La interpretación real de todo lo que se ha comentado es que, si fuéramos propietarios de esta empresa de autobuses donde se producen estos errores, el primer error en el que nos deberíamos centrar en solucionar sería el $\mathcal{C}_{E1}$. Este error se corresponde con cruzar un semáforo en ambar. Habrá distintas formas de enfocar este problema y de intentar encontrar una solución, como el hecho de realizar cursos de concienciación, intentar disminuir las horas que un conductor está conduciendo el autobús de continuo, etc. Se pueden probar distintos métodos, y una vez que la frecuencia con la que se produce este error disminuya de manera considerable, podemos intentar solucionar el siguiente evento, que en este caso es $\mathcal{C}_{E2}$. Se procedería así sucesivamente hasta conseguir que todos los eventos tengan un riesgo bajo o, si esto no es posible, conseguir disminuirlo de forma considerable.

B. *Ranking de Condorcet*

En muchas situaciones reales habituales es difícil que cuando obtenemos distintos rankings y el objetivo es obtener un ranking consenso se tenga un ranking que aparezca más de la mitad de las ocasiones. Para solucionar este problema, Condorcet se apoya en el concepto de ganador por mayoría. Condorcet propone un ganador basado en la dominancia del mismo sobre el resto de candidatos. Si tal candidato existe, será elegido el ganador.

Para ver cuál es el ganador de Condorcet (para más información, ver de nuevo [20]), debemos buscar cuál es el candidato que comparado con el resto es el preferido por el mayor número de votantes. En este ejemplo, deberemos mirar cuál es el evento que tiene mayor importancia en la mayoría de los órdenes lexicográficos.

Se puede observar que el evento $\mathcal{C}_{E1}$ es elegido como primero un total de 16 veces sobre 24 posibles clasificaciones. Como 16 es más de la mitad del número de ordenaciones, este candidato será también preferido por pares a todos los demás.

Además en este caso, el único evento que le hace una competencia real a $\mathcal{C}_{E1}$ es $\mathcal{C}_{E2}$, ya que este es el único que le supera en alguna ocasión (lo hace en las 8 restantes). Así el primer evento sobre el que deberíamos prestar atención es el C_{E1}. Para ver qué sucede con el resto de la clasificación, debemos seguir mirando qué sucede con el resto de pares. El segundo evento sobre el que nos deberíamos fijar es sobre $\mathcal{C}_{E2}$ ya que domina al resto de errores (salvo $\mathcal{C}_{E1}$, como hemos mencionado anteriormente) de manera absoluta. De igual manera $\mathcal{C}_{E4}$ domina al resto de los eventos (salvo a $\mathcal{C}_{E1}$ y a $\mathcal{C}_{E2}$), estamos en la misma situación con los eventos $\mathcal{C}_{E3}$ y $\mathcal{C}_{E5}$. Todos ellos tienen dominancia sobre los eventos siguientes, $\mathcal{C}_{E3}$ tiene dominancia sobre $\mathcal{C}_{E5}$, $\mathcal{C}_{E6}$ y $\mathcal{C}_{E7}$ y $\mathcal{C}_{E5}$ tiene dominancia sobre los eventos $\mathcal{C}_{E6}$ y $\mathcal{C}_{E7}$. Finalmente, $\mathcal{C}_{E7}$ tiene dominancia sobre el evento $\mathcal{C}_{E6}$.

Así, la clasificación según Condorcet quedaría de la siguiente manera: $\mathcal{C}_{E6} \leq \mathcal{C}_{E7} \leq \mathcal{C}_{E5} \leq \mathcal{C}_{E3} \leq \mathcal{C}_{E4} \leq \mathcal{C}_{E2} \leq \mathcal{C}_{E1}$. En este caso, la clasificación de Borda y de Condorcet coinciden, lo cuál ya habíamos comentado que suele ser habitual. Como ambos criterios coinciden, tanto el diagrama de Hasse, como la interpretación que podemos hacer en este caso es exactamente la misma que ocurría cuando utilizábamos el método de clasificación de Borda. Nótese que el método de Condorcet no siempre da lugar a una ordenación de los eventos (puede haber ciclos).

C. Orden de Xu-Yager

Por otro lado, si consideramos la generalización del orden de Xu-Yager, obtenemos la siguiente clasificación: $\mathcal{C}_{E1} \geq_{XY} \mathcal{C}_{E2} \geq_{XY} \mathcal{C}_{E4} \geq_{XY} \mathcal{C}_{E3} \geq_{XY} \mathcal{C}_{E5} \geq_{XY} \mathcal{C}_{E7} \geq_{XY} \mathcal{C}_{E6}$.

En este caso, sucede lo contrario que cuando estudiamos los órdenes lexicográficos, tenemos una única clasificación posible para todos ellos, ya que los dos primeros puntos de la definición de $\leq_{XY}$ son suficientes para determinar la clasificación.

Las conclusiones según este orden coincide que se corresponden con las dadas por la fusión de los órdenes lexicográficos, tanto aplicando Borda, como Concordet.

V. Conclusiones

Hemos introducido un concepto que generaliza al de intervalo y nos permite definir una matriz de riesgo bajo imprecisión, además de analizar distintas estructuras de orden sobre el conjunto de cajas. La nueva matriz generaliza a la usual, manteniendo coherencia con el significado de las etiquetas y los colores de la misma, pero permitiendo además hacer consideraciones menos estrictas. Los elementos de dicha matriz pueden ser ordenados con distintos órdenes totales y en un ejemplo aplicado vemos como dicha información puede ser fusionada aplicando los métodos de Borda y Concordet, que en este caso llegan a la misma conclusión.

Es inmediato observar que las cuestiones estudiadas plantean la necesidad de resolver algunos problemas bastante directos. Por ejemplo, considerar otro orden en la definición de caja, estudiar otros órdenes en el conjunto de cajas o utilizar otros métodos de elecciones para fusionar la ordenación dada por distintos criterios.

Referencias

[1] O.N. Aneziris, I.A. Papazoglou, M.L. Mud, M. Damen, J. Kuiper, H. Baksteen, B.J. Ale, L.J. Bellamy, A.R. Hale, A. Bloemhoff, J.G. Post, and J. Oh. Towards risk assessment for crane activities. *Safety Science*, 46:872–884, 2008.

[2] G.E. Apostolakis, V.M. Bier, and A. Mosleh. A critique of recent models for human error rate assessment. *Reliab. Eng. Syst. Saf.*, 22:201–217, 1988.

[3] B. Bedregal, G. Beliakov, H. Bustince, T. Calvo, R. Mesiar, and D. Paternain. A class of fuzzy multisets with a fixed number of memberships. *Information Sciences*, 189:1—17, 2012.

[4] L.J. Bellamy, T.A.W. Geyer, and J.A. Astley. Evaluation of the human contribution to pipework and in-line equipment failure frequencies. Technical Report 15, HSE Contract Research Report, UK Health and Safety Executive, Bootle, Merseyside, 1989.

[5] K. Bogart, J. Bonin, and J. Mitas. Interval orders based on weak orders. *Discrete Applied Mathematics*, 60(1):93–98, 1995.

[6] H. Bustince, J. Fernandez, A. Kolesárová, and R. Mesiar. Generation of linear orders for intervals by means of aggregation functions. *Fuzzy Sets and Systems*, 220:69 – 77, 2013.

[7] E. Calixto, G.B.A. Lima, and P.R.A. Firmino. Comparing slim, spar-h and bayesian network methodologies. *Open Journal of Safety Science and Technology*, 3(2):31–41, 2013.

[8] F. Castiglia and M. Giardina. Analysis of operator human errors in hydrogen refuelling stations: comparison between human rate assessment techniques. *Int. J. Hydrog. Energy*, 38:1166–1176, 2013.

[9] L.A. Cox. What's wrong with risk matrices? *Risk Analysis*, 28:497—512, 2008.

[10] H.S.M. Coxeter. *Regular Polytopes*. Dover, New York, 1973.

[11] L. De Miguel, M. Sesma-Sara, M. Elkano, M. Asiain, and H. Bustince. An algorithm for group decision making using n-dimensional fuzzy sets, admissible orders and owa operators. *Information Fusion*, 37:126–131, 2017.

[12] G. Desmorat, F. Guarnieri, D. Besnard, P. Desideri, and F. Loth. Pouring cream into natural gas: the introduction of common performance conditions into the safety management of gas networks. *Saf. Sci.*, 54:1–7, 2013.

[13] P. Fishburn. Interval ordenings. *Wiley, New York*, 1987.

[14] J. Forester, D. Bley, S. Cooper, E. Lois, N. Siu, A. Kolaczkowski, and J. Wreathall. Expert elicitation approach for performing atheana quantification. *Reliability Engineering & System Safety*, 83(2):207–220, 2004.

[15] J.A Goguen. L-fuzzy sets. *Journal of Mathematical Analysis and Applications*, 18(1):145 – 174, 1967.

[16] K.M. Groth and L.P. Swiler. Bridging the gap between hra research and hra practice: a bayesian network version of spar-h. *Reliab. Eng. Syst. Saf.*, 115:33–42, 2013.

[17] L. Hurwicz. A class of criteria for decision-making under ignorance. *Cowles Comision Discussion Paper: Statistics 356*, 1951.

[18] G.K. Kaya. *Good risk assessment practice in hospitals*. PhD thesis, University of Cambridge, 2018.

[19] A.S. Markowski and M.S. Mannan. Fuzzy risk matrix. *Journal of hazardous materials*, 159(1):152—157, November 2008.

[20] R. Pérez-Fernández and B. De Baets. The superdominance relation, the positional winner, and more missing links between borda and condorcet. *Journal of Theoretical Politics*, 31:46—65, 2021.

[21] J.K. Satia and R.E. Lave. Markovian decision processes with uncertain transition probabilities. *Operations Research*, 21(3):728–740, 1973.

[22] M. Sniedovich. Wald's maximin model: a treasure in disguise! *Journal of Risk Finance*, 9(3):287–291, 2008.

[23] Virtual Scientific Meeting EUREKA 2016. *Transforming Risk Assessment Matrices via Receiver Operating Characteristic Curves*, Coahuila, México, 12 2016.

[24] A. Wald. Statistical decision functions which minimize the maximum risk. *Annals of Mathematics*, 46(2):265–280, 1945.

[25] Z.S. Xu and R.R. Yager. Some geometric aggregation operators based on intuitionistic fuzzy sets. *International Journal of General Systems*, 35(4):417–433, 2006.

[26] B. Zimolong. Empirical evaluation of therp, slim and ranking to estimate heps. *Reliab. Eng. Syst. Saf.*, 35:1–11, 1992.

Consenso en Entornos Académicos Virtuales: Toma de Decisiones Grupales en el Trabajo de Clase Online

1st Aitor Manuel Orellana Romero
Dept. de Ingeniería Informática
Universidad de Cádiz
Cádiz, España
aitormanuel.oreromero@alum.uca.es

2nd Antonio Velez-Estevez
Dept. de Ingeniería Informática
Universidad de Cádiz
Cádiz, España
antonio.velez@uca.es

3rd Manuel Jesús Cobo Martín
Dept. de Ingeniería Informática
Universidad de Cádiz
Cádiz, España
manueljesus.cobo@uca.es

4th Ignacio Javier Pérez Gálvez
Dept. de Ingeniería Informática
Universidad de Cádiz
Cádiz, España
ignaciojavier.perez@uca.es

***Abstract*—En esta contribución se describe la implementación de una plataforma web que puede ser integrada en el campus virtual de cualquier centro educativo con el objetivo de centralizar la gestión de las tareas de toma de decisiones en grupo por parte de los estudiantes. Debido a la situación actual propiciada por el virus COVID-19, muchas clases que antes eran presenciales se están impartiendo de forma "online". Por tanto, situaciones de toma de decisiones relativas al trabajo en grupo que hasta hace poco se resolvían directamente con un rápido debate en clase, ahora necesitan llevarse a cabo de forma no presencial. Dicha plataforma actúa como moderador virtual, permitiendo a los usuarios comunicarse a través de la web con el objetivo de tomar decisiones consensuadas.**

***Index Terms*—Toma de decisiones en grupo, Consenso, Lógica difusa, Clase online**

I. Introducción

La toma de decisiones es una tarea cotidiana que todos realizamos a diario. Hay decisiones con poca trascendencia como la de escoger si tomar café o zumo en el desayuno, y otras más relevantes como decidir si cambio de trabajo o mantengo mi puesto actual. La actividad docente no está excluida de este tipo de tareas, tanto desde el punto de vista del profesor como desde las tareas intrínsecas del estudiante [14]. Decidir qué día de la semana hacer una tarea, la fecha de un examen o escoger el tema de un trabajo de entre un conjunto de posibilidades, son decisiones que se toman frecuentemente en la universidad.

Asimismo, cuando se trabaja en grupo y la decisión afecta a varias personas, la situación empieza a ser más compleja. Este caso es conocido como el problema de la toma de decisiones en grupo. Resolver este problema de forma presencial suele ser relativamente sencillo; se discuten conjuntamente las diferentes alternativas para intentar llegar a una solución conjunta que sea "lo mejor para todos". Sin embargo, analizándolo bien, se pueden dar situaciones en las que cada persona perteneciente al grupo tenga unos intereses o motivaciones diferentes, incluso que dé más importancia a unos criterios de elección diferentes al resto de compañeros [9].

Cuando los alumnos se encuentran en una de estas situaciones, sumando a la ecuación que el entorno de estudio no sea presencial, necesitan herramientas o entornos virtuales que les ayuden con el proceso de toma de decisiones en grupo, de forma que no se paralice el trabajo sencillamente porque no hay consenso en alguna decisión que afecta al grupo en su conjunto. Por el momento, los estudiantes disponen de herramientas que les facilitan la comunicación, como foros, chats, o incluso wikis que fomentan el trabajo colaborativo. Sin embargo, no tienen ningún soporte para ayudarles a llegar a acuerdos grupales.

Cuando hablamos de toma de decisiones en grupo, hay varias formas de llegar a un acuerdo [5]. Desde un punto de vista más dominante, en el que habría un líder que toma la decisión y todos los demás le hacen caso (casuística poco recomendable en el entorno docente), hasta modelos más participativos en los que se hace lo que dice la mayoría (usando un sistema de votación en el que la alternativa que tenga más votos será la elegida como decisión del grupo). Sin embargo, tanto la primera como la segunda opción pueden dejar a algunos miembros del equipo sin sentirse cómodos con la decisión tomada, creando diferentes bandos en el grupo de trabajo. Lo ideal sería alcanzar algún tipo de consenso [2].

Dicho esto, el objetivo general de este proyecto es el de implementar un modelo de consenso usando la lógica difusa [1], [11], [12], [22], en el que todos los miembros del grupo se sientan igualmente representados por la decisión tomada. Para ello, no trataremos el concepto de consenso como un concepto "crisp" binario (hay consenso total o no lo hay), si no que se establecerá una medida del nivel de consenso alcanzado por el grupo en base a sus preferencias individuales (valores entre

cero y uno), y se establecerá un umbral para considerar si hay consenso suficiente o la negociación debe continuar [2], [16], [17]. De esta forma, usamos es término "Soft consensus" para referirnos a que el concepto de consenso puede tener diferentes grados y para poder medir el nivel de consenso que existe en cada momento en los procesos de toma de decisiones.

Para guiar la negociación de forma que el proceso sea convergente (que cada vez las posiciones de los estudiantes estén más cerca), se establecerá un mecanismo de consenso, que actuará como moderador virtual enviando un mensaje personalizado a cada uno de los miembros del grupo para que relaje un poco sus preferencias individuales en la dirección que marque el grupo en su conjunto.

De esta forma, tendremos dos procesos diferenciados que actúan conjuntamente de forma secuencial. El proceso de consenso actúa para lograr alcanzar el máximo grado de consenso posible entre las opiniones de los usuarios. Cuando todos han expresado sus opiniones, el sistema calcula el grado de consenso existente. Si es satisfactorio, entonces se aplica el proceso de selección de cara a obtener la solución final. Por el contrario, si el grado de consenso medio no es satisfactorio, entonces el sistema insta a los usuarios a modificar sus opiniones de cara a aumentar la proximidad en sus preferencias. De esta manera, un proceso de toma de decisiones puede verse como un proceso dinámico e iterativo en el que los usuarios van acercado sus posiciones hasta maximizar el consenso.

En la plataforma quedarán registradas todas las interacciones de los alumnos con el sistema, de forma que el profesor dispondrá de nuevos indicadores para la evaluación de la competencia de trabajo en grupo en caso de ser evaluable según el plan docente de cada asignatura [6], [15], [18].

El resto de este documento se estructura de la siguiente forma. La Sección 2 muestra el estado del arte. En la Sección 3 se describe la plataforma implementada. Finalmente, la Sección 4 presenta las conclusiones obtenidas y los trabajos futuros.

II. Estado del Arte

En esta sección vamos a mostrar los conceptos técnicos en los que se basa el modelo implementado.

A. Conceptos Básicos

Formalmente, el problema subyacente a un proceso de toma de decisiones en grupo se puede definir de la siguiente manera:

Sea $X = \{x_1, x_2, \ldots, x_n\} (n \geq 1)$ un conjunto de alternativas posibles y, teniendo en cuenta los valores de preferencia, $P = \{p_1, \ldots, p_m\}$, proporcionados por un grupo de expertos $E = \{e_1, \ldots, e_m\}$, ¿cómo deben ordenarse los valores del conjunto X de mejor a peor alternativa posible?

Por lo general, para resolver el problema, los procesos de toma de decisiones en grupo siguen los siguientes pasos [7]:

1) **Introducción de preferencias en el sistema**: Los expertos proporcionan sus preferencias al sistema. Las preferencias definen directa o indirectamente un orden sobre el conjunto de alternativas.
2) **Cálculo de la matriz colectiva de preferencias**: La información de las preferencias proporcionadas por todos los usuarios es agregada en una sola pieza de información. La matriz colectiva representa la media de las preferencias proporcionadas.
3) **Proceso de selección de alternativas**: Usando la matriz colectiva y los operadores de selección deseados, se genera el ranking final de las alternativas.

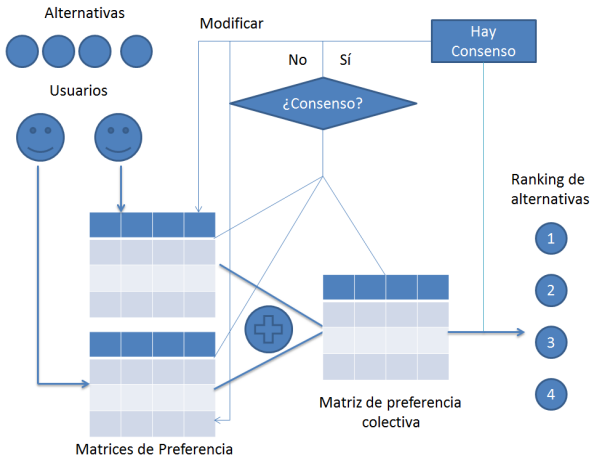


Fig. 1. Proceso de toma de decisiones con medidas de consenso.

El esquema comentado arriba tiene la desventaja de que no permite a los usuarios debatir ni llegar a ningún consenso antes de tomar la decisión final. Para solucionar este problema se utilizan las *medidas de consenso* [3]. Usando las matrices de preferencia de los expertos involucrados en el proceso de decisión, las medidas de consenso permiten determinar si los expertos opinan de forma parecida o si, por el contrario, tienen opiniones encontradas. De esta forma, si los expertos no llegan a un consenso, se les puede permitir que hablen y modifiquen sus preferencias con el objetivo de que se pongan de acuerdo. Si, por el contrario, todos están de acuerdo, se calcula el ranking de alternativas y el proceso de decisión termina. En la Fig. 1, podemos ver un esquema de como se definiría un proceso de toma de decisiones con medidas de consenso.

En un proceso de toma de decisiones, los expertos pueden proporcionar sus preferencias de diferentes formas. El procedimiento elegido es muy importante ya que establecerá la forma en que se deben realizar las operaciones necesarias para la toma de decisiones. Los métodos más comunes en la literatura son los siguientes [4], [19]:

- **Órdenes de preferencia**: El experto e_k proporciona sus preferencias utilizando una lista ordenada de preferencias $O^k = \{o^k(1), \ldots, o^k(n)\}$ donde $o^k(\cdot)$ se define como una función de permutación sobre el conjunto de índices $\{1, \ldots, n\}$ del conjunto de alternativas. De esta forma, las alternativas aparecen ordenadas de mejor a peor opción.

- **Funciones de utilidad**: El experto e_k comunica sus preferencias representadas como un conjunto de n valores de utilidad $U^k = \{u_i^k, i = 1, \ldots, n\}$, $u_i^k \in [0, 1]$ donde u_i^k representa la evaluación que el experto e_k proporciona a la alternativa x_i.
- **Relaciones de preferencia difusa**: El experto e_k proporciona sus preferencias mediante una relación $P^k \subset X \times X$ cuya función de pertenencia es $\mu_{P^k} : X \times X \rightarrow [0, 1]$. $\mu_{P^k} = p_{ij}^k$ establece el grado de preferencia de la alternativa x_i sobre x_j.

En este trabajo, vamos a utilizar las relaciones de preferencia difusas como formato de representación de preferencias.

B. Medidas de consenso y proximidad

Para calcular el consenso de un proceso de toma de decisiones que utiliza relaciones de preferencia difusa, podemos seguir los pasos expuestos en el artículo de Mata et al. [13] y que detallamos a continuación:

1) Para cada par de expertos e_i y e_j, calculamos las matrices de similaridad sm_{ij}. Para ello, aplicamos la siguiente función de similaridad para cada uno de los valores de preferencia de cada dos expertos:
$$s(p_i^{lk}, p_j^{lk}) = 1 - |(p_i^{lk} - p_j^{lk})| \quad (1)$$
donde $s(p_i^{lk}, p_j^{lk})$ muestra la similaridad entre las preferencias de las alternativas x_l sobre x_k para los expertos e_i y e_j.
2) Una vez calculadas todas las matrices se agregan en una única matriz de consenso colectiva. Para ello podemos utilizar el operador de media:
$$sm_c = \phi(sm_{ij}), \forall i, \forall j, i \neq j, i < j \quad (2)$$
3) Utilizando la matriz de consenso colectiva sm_c, podemos calcular tres medidas distintas de consenso, cada una representativa de un nivel diferente:
   a) *Nivel 1, consenso entre pares de alternativas*: Cada valor de la matriz sm_c nos muestra el consenso alcanzado para cada par de alternativas:
   $$cp^{lk} = cm^{lk} \forall l, k = 1, \ldots, n, \wedge l \neq k \quad (3)$$
   donde n es el número de alternativas del proceso de toma de decisiones.
   b) *Nivel 2, consenso en cada alternativa*: Para cada alternativa x_l, puede calcularse el nivel de consenso alcanzado, ca^l, usando la matriz cp tal y como muestra la siguiente expresión:
   $$ca^l = \frac{\sum_{k=1, l \neq k}^{n} (cp^{lk} + cp^{kl}))}{2(n-1)} \quad (4)$$
   c) *Nivel 3, consenso general del proceso*: Finalmente, podemos agregar los valores de consenso de cada una de las alternativas para obtener un valor de consenso global:
   $$cr = \sum_{l=1}^{n} ca^l / n \quad (5)$$

También es interesante calcular la distancia que hay entre las preferencias de cada uno de los expertos a la matriz colectiva global. De esta forma, podemos ver si las opiniones del experto son similares o no a la de los demás y en que grado. Estas medidas de proximidad [10], al igual que las de consenso, se pueden calcular en tres niveles distintos:

1) *Nivel 1, proximidad en cada par de alternativas*: El nivel de proximidad para cada par de alternativas (x_l, x_k), pp_i, del experto e_1, puede calcularse de la siguiente forma:
$$pp_i^{lk} = s(p_i^{lk}, p_c^{lk}) \quad (6)$$
donde p_c es la matriz colectiva.
2) *Nivel 2, proximidad para cada alternativa*: De manera análoga que en el consenso, podemos calcular el nivel de proximidad del experto a cada una de las alternativas mediante la siguiente expresión:
$$pa_i^l = \frac{\sum_{k=1, l \neq k}^{n} (pp_i^{lk} + pp_i^{kl})}{2 \cdot (n-1)} \quad (7)$$
3) *Nivel 3, Proximidad general*: El nivel de proximidad general de las preferencias del experto e_i puede calcularse usando la siguiente expresión:
$$pr_i = \frac{pa_i^l}{n} \quad (8)$$

Estas medidas serán útiles a la hora de identificar aquellos usuarios mas alejados de la opinión colectiva del grupo, y que por tanto, deberían acercar posturas con el mismo para maximizar el grado de consenso.

C. Métodos de agregación de información

Para calcular la matriz colectiva de preferencias es necesario agregar la información proporcionada por los expertos. Para ello debemos usar algún operador de agregación. A continuación expondremos algunos operadores que pueden usarse para completar esta tarea:

- el operador de media.
- el de media ponderada.
- el operador de media de pesos ordenados (OWA) [20], [21].

Para calcular la matriz de preferencias colectiva utilizando el operador de media podemos utilizar la siguiente expresión:
$$C_{ij} = \frac{p_{ij}^1 + \ldots + p_{ij}^n}{m} \quad (9)$$

D. Operadores de selección

Para el proceso de selección, se utilizan los operadores de selección. Este tipo de operadores son capaces de obtener un ranking de alternativas a partir de una matriz colectiva de preferencias. Dos ejemplos de este tipo de operadores son los operadores de dominancia y no dominancia, GDD y GNDD respectivamente [8]. El operador GDD calcula el grado en que una alternativa domina a otra mientras que el de no dominancia se encarga de determinar qué alternativas no son dominadas por otras.

El operador GDD se calcula mediante la siguiente expresión:

$$GDD_i = \phi(c_{i1}, c_{i2}, \ldots, c_{i(i-1)}, c_{i(i+1)}, \ldots, c_{in}) \quad (10)$$

donde c es la matriz de preferencia colectiva y ϕ representa el operador de media.

El operador GNDD puede calcularse utilizando la siguiente expresión:

$$GNDD_i = \phi(c^s_{1i}, c^s_{2i}, \ldots, c^s_{(i-1)i}, c^s_{(i+1)i}, \ldots, c^s_{ni}) \quad (11)$$

donde

$$c^s_{ji} = \max\{c_{ji} - c_{ij}, 1\}$$

III. Plataforma de Ayuda a la Toma de Decisiones Grupales en el Trabajo de Clase Online

En esta sección vamos a mostrar el funcionamiento de la plataforma desarrollada.

A. *Tecnologías empleadas para el desarrollo de la plataforma*

A nivel tecnológico, la plataforma ha sido desarrollada teniendo como cimiento de programación el lenguaje PHP 8 y, a mas alto nivel, se ha utilizado el framework de aplicaciones web PHP Laravel.

Dicha plataforma cuenta entre sus dependencias con una extensión de seguridad diseñada por el propio ámbito de Laravel llamado Laravel JetStream. Dicha extensión nos ofrece una funcionalidad complementaria para la autenticación de usuarios y todo el conjunto referente a su gestión dentro de la plataforma. Incluyéndose el uso de los "teams" para los roles de los usuarios.

Para el envío de emails a los usuarios con la información sobre las decisiones a tomar o ya tomadas, se ha utilizado la dependencia PHPMailer por un servidor SMTP (Simple Mail Transfer Protocol) con un correo previamente definido y adecuado para esta envergadura.

El estilo de la página viene predefinido por una base de Boostrap 5 con modificaciones e inclusión de clases de CSS propias.

La plataforma se basa en un modelo de tres capas, definida como Modelo-Vista-Controlador (MVC). La base de datos que utiliza el modelo es una base de datos relacional MySQL.

B. *Ejemplo de uso y funcionalidad ofrecida por la Plataforma*

En la pantalla inicial (ver Fig. 2), se puede configurar un nuevo proceso de decisión en grupo ajustando las alternativas y los expertos, que son divididos en diferentes roles según los tipos de decisiones que van a tomar. Una vez configurado el proceso, los usuarios pueden proceder a insertar sus preferencias individuales (ver Fig.3) .

Cuando el usuario se identifica en la plataforma, tendrá una sección llamada "Mis Participaciones" donde puede ver todos los procesos de decisión pendientes en los que se ha visto involucrado y el estado en el que se encuentran.

Otra de las operaciones que ofrece la plataforma es el panel de administración (ver Fig.4). El administrador de cada



Fig. 2. Pantalla inicial

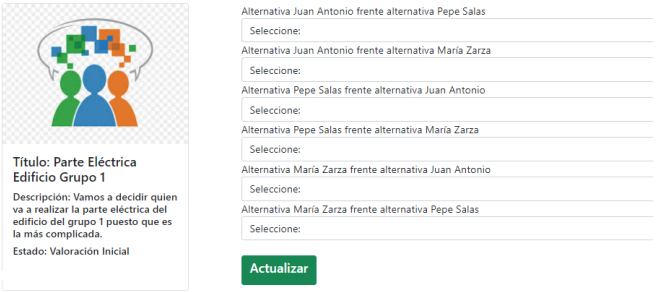


Fig. 3. Inserción de preferencias sobre las alternativas

proceso podrá ver qué usuarios han registrado sus preferencias, el estado de la decisión y en el caso de que el nivel de consenso sea suficiente, la alternativa escogida como mejor opción.



Fig. 4. Panel de Administración

Una vez que todos los usuarios han insertado sus preferencias, al administrador se le enviará un mensaje con los resultados obtenidos (ver Fig.5) y, en caso de no alcanzar el nivel de consenso requerido, tendrá la opción de iniciar una nueva ronda de valoraciones de preferencias, mostrando previamente a cada experto, no solo el consenso alcanzado a nivel de alternativas, sino también la proximidad con respecto a la opinión colectiva para que vea si es necesario actualizar sus preferencias (y en qué dirección) como mecanismo de retroalimentación que trate de conseguir que el proceso sea convergente y en cada iteración el nivel de consenso sea mayor que en la anterior.

En cambio, si la acción escogida es "Terminar Valoración",

debido a que el nivel de consenso ya es satisfactorio, el proyecto pasará a un estado de "Valoración Final" donde ya los expertos tendrán en el apartado "Decisiones Tomadas" un mensaje con el resultado sobre la decisión final consensuada.



Fig. 5. Estado actual del proceso de toma de decisiones

Finalmente, en la Fig. 6, se puede observar la versión de la aplicación móvil de la plataforma funcionando bajo el sistema operativo Android.

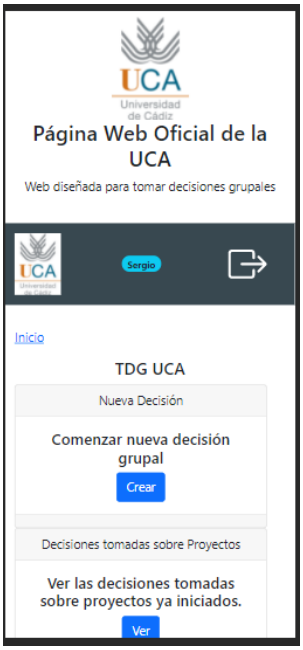


Fig. 6. Aplicación Android

IV. Conclusiones y trabajos futuros

En esta contribución se ha presentado una plataforma web de toma de decisiones en grupo, orientada al caso práctico del trabajo grupal en clase en un entorno docente no presencial propiciado por el Covid-19. De esta manera, la plataforma creada actúa como moderador virtual, ayudando a tomar decisiones sensatas para todos los participantes.

Como trabajos futuros, se propone la extensión de la plataforma para aceptar diferentes estructuras de representación de preferencias. También se plantea la posibilidad de implementar un sistema de control de consistencia y de estimación de preferencias no rellenas. Por último, se desea implementar una plataforma dinámica, que permita que los elementos del problema puedan cambiar una vez comenzado el proceso de negociación.

Agradecimientos

Este trabajo ha sido financiado por la Agencia Estatal de Investigación a través del proyecto "iScience" PID2019-105381GA-I00 / AEI / 10.13039/501100011033 , por el Programa de Proyectos de Innovación y Mejora Docente de la Universidad de Cádiz (UCA/R100REC/2020) y por el Programa de Fomento e Impulso de la actividad Investigadora de la Universidad de Cádiz.

Referencias

[1] Bellman, R.E., Zadeh, L.A.: Decision-making in a fuzzy environment. Management science **17**(4), B–141 (1970)
[2] Cabrerizo, F.J., Moreno, J.M., Pérez, I.J., Herrera-Viedma, E.: Analyzing consensus approaches in fuzzy group decision making: advantages and drawbacks. Soft Computing **14**(5), 451–463 (2010)
[3] Cabrerizo, F.J., Moreno, J.M., Pérez, I.J., Herrera-Viedma, E.: Analyzing consensus approaches in fuzzy group decision making: advantages and drawbacks. Soft Computing **14**(5), 451–463 (2010)
[4] Chiclana, F., Herrera, F., Herrera-Viedma, E.: Integrating three representation models in fuzzy multipurpose decision making based on fuzzy preference relations. Fuzzy sets and Systems **97**(1), 33–48 (1998)
[5] Collins, B.E., Guetzkow, H.S.: A social psychology of group processes for decision-making. Wiley New York (1964)
[6] García-Cabrero, B., Luna-Serrano, E., Ponce-Ceballos, S., Cisneros-Cohenour, E., Cordero-Arroyo, G., Espinoza-Díaz, Y., García-Vigil, M.H.: Las competencias docentes en entornos virtuales: Un modelo para su evaluación. Revista Iberoamericana de Educación a Distancia **21**(1), 343–365 (2018)
[7] Herrera, F., Alonso, S., Chiclana, F., Herrera-Viedma, E.: Computing with words in decision making: foundations, trends and prospects. Fuzzy Optimization and Decision Making **8**(4), 337–364 (2009)
[8] Herrera, F., Herrera-Viedma, E., Verdegay, J.L.: A sequential selection process in group decision making with a linguistic assessment approach. Information Sciences **85**(4), 223–239 (1995)
[9] Herrera-Viedma, E., Cabrerizo, F.J., Kacprzyk, J., Pedrycz, W.: A review of soft consensus models in a fuzzy environment. Information Fusion **17**, 4–13 (2014)
[10] Herrera-Viedma, E., Martinez, L., Mata, F., Chiclana, F.: A consensus support system model for group decision-making problems with multigranular linguistic preference relations. Fuzzy Systems, IEEE Transactions on **13**(5), 644–658 (2005)
[11] Huang, C.L., Lo, C.C., Chao, K.M., Younas, M.: Reaching consensus: A moderated fuzzy web services discovery method. Information and Software Technology **48**(6), 410–423 (2006)
[12] Kacprzyk, J.: Group decision making with a fuzzy linguistic majority. Fuzzy sets and systems **18**(2), 105–118 (1986)
[13] Mata, F., Martínez, L., Herrera-Viedma, E.: An adaptive consensus support model for group decision-making problems in a multigranular fuzzy linguistic context. IEEE Transactions on fuzzy Systems **17**(2), 279–290 (2009)
[14] Morente-Molinera, J., Castellón-Fuentes, J., Herrera-Viedma, E., López-Herrera, A.: A decision making web platform for educational centers. In: INTED2013 Proceedings. pp. 1368–1376. IATED (2013)
[15] Pazmiño, D.X.S.: Evaluación de las competencias digitales en la formación inicial docente de los y las estudiantes de la Facultad de Filosofía, Letras y Ciencias de la Educación de la Universidad Central del Ecuador. Ph.D. thesis, Universitat d'Alacant-Universidad de Alicante (2021)
[16] Pérez, I.J., Cabrerizo, F.J., Alonso, S., Dong, Y., Chiclana, F., Herrera-Viedma, E.: On dynamic consensus processes in group decision making problems. Information Sciences **459**, 20–35 (2018)
[17] Pérez, I.J., Cabrerizo, F.J., Alonso, S., Herrera-Viedma, E.: A new consensus model for group decision making problems with non-homogeneous experts. IEEE Transactions on Systems, Man, and Cybernetics: Systems **44**(4), 494–498 (2013)

[18] Pueyo, Á.P., Berrocal, O.M.C., Bernardino, C.H., Martín, J.J.B., Cobo, D.V., González, L.P., Álvarez, I.H., Fernández, L.C., Benito, R.M., Nava, M.M., et al.: Programar y evaluar competencias básicas en 15 pasos, vol. 301. Graó (2013)
[19] Tanino, T.: On group decision making under fuzzy preferences. In: Multiperson Decision Making Models using Fuzzy Sets and Possibility Theory, pp. 172–185. Springer (1990)
[20] Yager, R.R.: On ordered weighted averaging aggregation operators in multicriteria decisionmaking. Systems, Man and Cybernetics, IEEE Transactions on **18**(1), 183–190 (1988)
[21] Yager, R.R.: Quantifier guided aggregation using OWA operators. International Journal of Intelligent Systems **11**(1), 49–73 (1996)
[22] Zadeh, L.A.: The concept of a linguistic variable and its application to approximate reasoning-I. Information sciences **8**(3), 199–249 (1975)

XX Congreso Español sobre Tecnologías y Lógica Fuzzy

Sesión especial:

Funciones de agregación y preagregación y sus aplicaciones



Mapas de características adicionales en redes neuronales profundas a partir de agregaciones basadas en orden

I. Dominguez-Catena, D. Paternain, M. Galar
Institute of Smart Cities
Universidad Pública de Navarra, Pamplona, España
{iris.dominguez, mikel.galar, daniel.paternain}@unavarra.es

***Resumen*—En los últimos años hemos observado avances enormes en el área del Aprendizaje Automático, especialmente a través del uso de Redes Neuronales Profundas. Uno de los ejemplos más relevantes es la clasificación de imágenes, donde las Redes Neuronales Convolucionales han demostrado ser una herramienta muy precisa. Aunque las funciones de agregación, como los operadores OWA, ya han sido empleadas en combinación con redes neuronales, en este trabajo proponemos y exploramos una nueva forma de integrar operadores OWA en redes neuronales. Para ello, introducimos los operadores OWA dentro de una nueva capa en una red neuronal convolucional. Realizamos varios experimentos introduciendo la capa en una red basada en VGG-13 y mostramos como la capa introduce nuevo conocimiento en la red.**

***Index Terms*—Redes Neuronales, Redes Neuronales Convolucionales, aprendizaje profundo, operadores OWA**

I. Introducción

Uno de los problemas más estudiados en aprendizaje automático es el de clasificación supervisada de imágenes [1], [2]. En este tipo de problemas intentamos desarrollar un modelo matemático que aprende sobre un conjunto de imágenes etiquetadas, y que después sea capaz de etiquetar nuevos ejemplos apropiadamente. Actualmente, la técnica más común es el uso de redes neuronales convolucionales (CNNs), el foco de atención de este trabajo.

Las medidas ponderadas ordenadas (OWA) [3], [4] son un tipo de agregación paramétrica comúnmente utilizada en el campo del aprendizaje automático y la lógica difusa. En trabajos previos, los operadores OWA han sido integrados en aprendizaje profundo principalmente como un método para combinar las salidas de diferentes clasificadores (ensembles) [5], [6], [7], [8]. También han sido empleados con resultados interesantes en las capas de pooling de las CNNs [9].

A diferencia de los trabajos anteriores, intentamos emplear operadores OWA en las capas internas de una CNN, con el objetivo de incrementar la información disponible para la siguiente capa, añadiendo muy pocos parámetros a la red. Nuestro objetivo es generar información sin coste en la red, obteniendo los mapas de características en un punto de la red y añadiendo información derivada que sería difícil de conseguir a través de operadores convolucionales normales. Para esto, proponemos implementar una capa de agregaciones OWA a nivel de canal, que aprenda los pesos de varios operadores junto con los parámetros de la red, y los aplique para añadir mapas de características virtuales a la información ya existente.

Para comprobar el funcionamiento de esta propuesta hemos considerado una arquitectura base de tipo VGG13 [10], y hemos insertado capas OWA en ella. Probamos diferentes configuraciones de la capa sobre los datasets de clasificación de imágenes CIFAR10 y CIFAR100 [11].

El resto del trabajo está organizado de la siguiente manera. La Sección II describe la literatura relevante a nuestra propuesta. La Sección III revisa algunos conceptos preliminares sobre OWA y CNN. Después, la Sección IV especifica nuestra metodología para la inserción de la capa OWA y explica el funcionamiento interno de esta. La Sección V presenta los experimentos que hemos diseñado para probar la capa y los detalles de implementación. A continuación, la Sección VI recopilamos los resultados experimentales y los analizamos. Finalmente, la Sección VII concluye este trabajo y propone algunas líneas de trabajo futuras.

II. Literatura relacionada

En la literatura se han explorado previamente varias maneras de combinar operadores OWA y redes neuronales [5], [6], [7], [8], [9]. La manera más habitual es emplear OWAs sobre la salida de las redes, agregando sus resultados [5], [6], [7], [8]. La otra técnica habitual es la sustitución de las agregaciones en las capas de pooling por OWAs [9].

El empleo de operaciones de agregación basadas en medidas difusas [12], como las integrales Choquet y Sugeno (de las cuales los operadores OWA son un caso particular), para agregar ensembles de redes neuronales ha sido estudiado en múltiples ocasiones recientemente [5], [6], [7], [8]. En estos sistemas, se entrenan una serie de clasificadores independientemente, y se emplean operadores de agregación sobre los resultados. En este paso final es donde se pueden emplear operadores OWA y otros operadores de agregación basados en medidas difusas.

Los operadores basados en medidas difusas también se han utilizado como operadores de reducción en las capas de

Este trabajo ha sido financiado por el Servicio de Investigación de la Universidad Pública de Navarra bajo el proyecto PJUPNA1926, y por el MICIN de España (PID2019-108392GB-I00 / AEI / 10.13039/501100011033).

pooling de CNNs [9]. En este caso, las agregaciones habituales empleadas en las capas de pooling se reemplazan directamente por otros operadores. La idea es obtener representaciones más fieles de la información original tras la reducción de dimensionalidad que se da en estas capas.

Finalmente, la inspiración principal para este trabajo se encuentra en [13], donde los autores proponen la creación de lo que ellos denominan una "Capa Difusa". Esta capa, insertada en diferentes puntos de una CNN, realiza seis operaciones OWA predeterminadas (máximo, mínimo, máximo suavizado, mínimo suavizado, media y un operador aleatorio) sobre los canales de la red, ordenados en base a una medida de la entropía de cada canal. Mientras que los autores aplicaban su método a un problema de segmentación de imágenes [14], experimentalmente hemos comprobado que intentar trasladar la técnica a problemas generales de clasificación de imágenes no obtenía buenos resultados. En nuestra opinión, en el caso de las redes empleadas para clasificación, existe demasiada información codificada en el orden de los mapas de características, que se pierde al aplicar los operadores OWA. Por tanto, diseñamos nuestra propuesta con la idea de aumentar la información en la red, en vez de reemplazarla, concatenando los nuevos mapas de características a los ya presentes. De esta manera, empleamos la salida de nuestra capa OWA como un complemento a la salida de las convoluciones estándar, proporcionando a las siguientes capas información que sería de otra forma difícil de obtener (información global a partir de las métricas del canal). Además, incluimos los pesos de los operadores OWA como parámetros de la red, aprendiéndolos en vez de mantenerlos fijos como en [13].

III. Preliminares

III-A. Operadores OWA

Los *operadores OWA* fueron propuestos inicialmente por Yager [3]. Estos operadores son mapeos $F\colon \mathcal{R}^n \to \mathcal{R}$ basados en una colección de pesos $W = [w_1, \ldots, w_n]$, con la condición de $w_i \in [0, 1]$ para todo $i = 1, \ldots, n$ y $\sum_{i=1}^{n} = 1$, y definidos como:

$$F(a_1, \ldots, a_n) = \sum_{j=1}^{n} w_j b_j \tag{1}$$

donde b_j representa el j-ésimo elemento más grande de a_i.

Algunos ejemplos notables de operadores OWA serían el `máximo` ($W = [1, 0, \ldots, 0]$), `mínimo` ($W = [0, \ldots, 0, 1]$), y la `media aritmética` ($W = \left[\frac{1}{n}, \ldots, \frac{1}{n}\right]$).

III-B. Redes Neuronales Convolucionales Profundas

Las redes neuronales convolucionales (CNNs) modifican la arquitectura habitual de las redes neuronales para especializarse en información espacial [15]. El uso más común, el procesamiento de imágenes, supone reconocer las relaciones 2D espaciales de la información de entrada, y emplear operaciones convolucionales que sólo toman en consideración píxeles vecinos de la imagen. Esto también supone perder algo de información de gran escala de la imagen, al sólo trabajarse con píxeles cercanos.

Otra importante característica de las CNNs son las capas de *pooling* [1]. Estas capas, como las convolucionales, reconocen la estructura espacial de las imágenes y mapas de características derivados, pero en vez de agregar mapas (como las convolucionales) operan sobre un solo canal, resumiendo la información por bloques y reduciendo el tamaño de cada mapa de características independientemente.

Algunas arquitecturas de CNN conocidas que han sido desarrolladas para clasificación de imágenes son LeNet [16], la familia VGG [10] y ResNet [2]. En este trabajo nos centraremos en VGG, pero nuestra metodología podría ser extrapolada a casi cualquier arquitectura CNN.

IV. Metodología

IV-A. Capa OWA

Nuestra capa OWA propuesta funcionará tomando una entrada de N imágenes, con una resolución de I filas por J columnas y C_{in} canales de profundidad (mapas de características de entrada), y agregando C_f nuevos canales a los originales, con $C_f \in [0, C_{in}]$. La salida será de N imágenes con la misma resolución $I \times J$, pero $C_{out} = C_{in} + C_f$ canales de profundidad, $C_{out} \geq C_{in}$. Para generar esos C_f nuevos mapas de características, aplicaremos C_f operadores OWA sobre los canales de entrada. Estos operadores OWA compartirán la misma función de ordenación, que utilizará métricas calculadas por canal para reordenarlos. Después, cada uno de estos operadores OWA generará un nuevo mapa de características como una combinación lineal de los canales ordenados, a partir de un vector de pesos propio. Estos vectores de pesos, uno por cada uno de los C_f OWAs aprendidos, se aprenderán y actualizarán como parámetros de la red. Profundizamos más sobre esto en la Sección IV-C. La arquitectura general de la capa se muestra en la Figura 1.

IV-B. Ordenación de canales

Al decidir trabajar por canal (y no por píxel), resulta vital definir precisamente la función de ordenación, esto es, en base a que métrica ordenaremos los canales. Dado un canal X de tamaño $I \times J$, consideramos las siguientes métricas:

- *Entropía del canal*. Empleamos la fórmula de entropía de Shannon [17] aplicada a los valores de todos los píxeles del canal,

$$H(X) = -\sum_{i=1}^{I}\sum_{j=1}^{J} x_{ij} \log x_{ij} \tag{2}$$

  Como esta función está diseñada para trabajar sobre vectores de elementos en el rango $x_{ij} \in [0, 1]$ con $\sum_{i=1}^{I}\sum_{j=1}^{J} x_{ij} = 1$, primero aplicamos la función *softmax* a la entrada X para normalizar el mapa de características,

$$\text{Softmax}(x_{kl}) = \frac{e^{x_{kl}}}{\sum_{i=1}^{I}\sum_{j=1}^{J} e^{x_{ij}}} \tag{3}$$

  Intuitivamente, podemos entender la entropía como una medida de desorden, de la cantidad de información

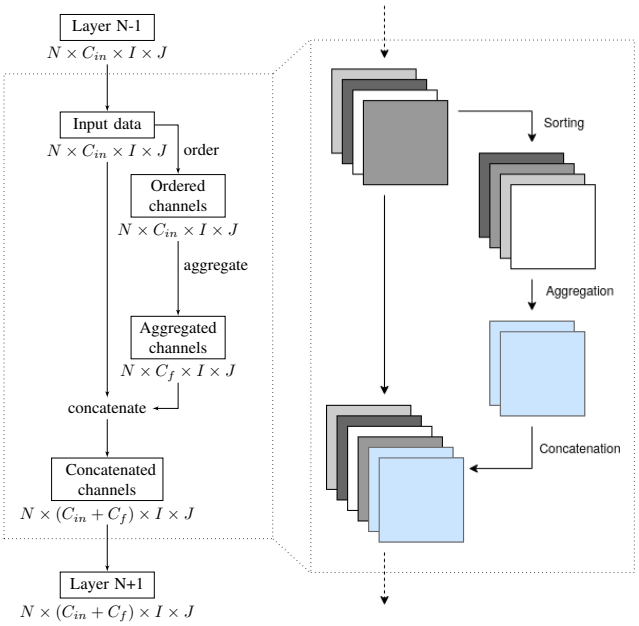


Figura 1. Propuesta de estructura para la capa OWA.

codificada en un canal. Un valor más elevado de entropía se corresponde con más uniformidad en los valores de entrada, mientras que un valor más pequeño se corresponde con un mayor contraste e información en la entrada.

- *Suma de valores*. Consideramos también la suma simple de activaciones en el canal,

$$S(X) = \sum_{i=1}^{I} \sum_{j=1}^{J} x_{ij} \tag{4}$$

- *Variación total* [18]. Considerando las características espaciales de la imagen, calculamos las diferencias entre cada píxel y su vecindad, tanto horizontal como verticalmente, y sumamos el valor absoluto de las diferencias en la imagen.

$$TV_v(X) = \sum_{i=2}^{I} \sum_{j=1}^{J} |x_{i,j} - x_{i-1,j}| \tag{5}$$

$$TV_h(X) = \sum_{i=1}^{I} \sum_{j=2}^{J} |x_{i,j} - x_{i,j-1}| \tag{6}$$

$$TV(X) = TV_v(X) + TV_h(X) \tag{7}$$

La Variación Total (TV), como se define en [18], es una medida que nos dice cuánta variación existe entre los píxeles de una imagen y sus vecinos. Esta medida será elevada para imágenes con muchos bordes nítidos y contraste elevado, y baja para imágenes muy uniformes.

- *Mediana de los valores*. Un operador OWA clásico,

$$M(X) = median(x_{11}, \ldots, x_{IJ}) \tag{8}$$

donde el operador mediana devuelve el $ceil(I \cdot J/2)$-ésimo elemento más grande de X si $I \cdot J$ es impar o la media aritmética de los $I \cdot J/2$-ésimo y $I \cdot J/2+1$-ésimos elementos más grandes de X, si $I \cdot J$ es par.

- *Máximo de los valores*. En este contexto, el valor del píxel más activado del canal,

$$MAX(X) = \text{máx}(x_{11}, \ldots, x_{IJ}) \tag{9}$$

Hemos considerado, además, dos métodos de ordenación de referencia que no se basan en los valores del canal:

- *No ordenación*. Mantenemos los canales en el orden que vienen, convirtiendo el OWA en una agregación estándar.
- *Ordenación aleatoria*. Esta ordenación nos permitirá introducir ruido equivalente al resultado de aplicar la capa OWA, permitiéndonos descartar que se obtengan mejoras en la precisión debido a fenómenos de tipo regularización.

IV-C. Agregación ponderada

Con los canales de entrada ya ordenados, realizaremos una agregación ponderada, empleando C_f vectores de pesos (uno por OWA), cada uno de ellos con un peso por cada canal de entrada (C_{in} en total). En nuestra propuesta, los inicializaremos de forma aleatoria siguiendo una distribución $U(0, 1)$. Trataremos estos pesos como parámetros de la red, y por tanto se aprenderán a través del método de retropropagación habitual.

Estos pesos no se encuentran directamente restringidos, y conforme son aprendidos pueden llegar a tomar cualquier valor. Para ajustarnos a la definición de OWA dada en la Sección III-A, donde para cada $w_i, i \in 1, \ldots, C_f$, requerimos $w_i \in [0, 1]$ y $\sum_{i=1}^{C_f} w_i = 1$, aplicamos algunas transformaciones antes de realizar la agregación. Primero, empleamos una función ReLU para convertir los pesos negativos en 0, y después normalizamos dividiendo el vector por la suma de sus valores, de forma que sumen en total 1.

$$ReLU(x) = max(x, 0) \tag{10}$$

$$w_j = \frac{ReLU(x_j)}{\sum_{i=1}^{C_f} ReLU(x_i)} \tag{11}$$

El resultado es un vector de pesos OWA correcto que puede ser aplicado directamente a los canales.

V. Marco experimental

V-A. Datasets

Como datasets de pruebas hemos escogido CIFAR10 y CIFAR100 [11]. Ambos son datasets de clasificación de imágenes muy conocidos, compuestos cada uno por 60.000 imágenes en color en una resolución de 32x32 píxeles. En el caso de CIFAR10 se reparten en 10 clases balanceadas y fácilmente distinguibles (6.000 por clase), mientras que CIFAR100 etiqueta sus imágenes en 100 clases (600 ejemplos por clase). Ambos distribuyen las imágenes en un conjunto de entrenamiento de 50.000 imágenes y un conjunto de prueba de 10.000, balanceados por clase.

La elección de estos datasets está motivada por el número de pruebas de configuraciones diferentes que queremos realizar. Estos datasets de pequeño tamaño nos facilitan realizar múltiples repeticiones y entrenamientos por cada configuración, de manera que podamos obtener resultados promediados estables.

V-B. Arquitectura

Decidimos trabajar con una arquitectura de referencia de tipo VGG [10]. Se trata de una arquitectura relativamente simple pero con muy buenos resultados, y el entrenamiento es lo suficientemente rápido como para realizar múltiples repeticiones. Al tener una estructura lineal, nos ofrece múltiples puntos de inserción para nuestra capa OWA sobre los que evaluar sus características.

De las posibles configuraciones de esta familia hemos seleccionado la VGG13, que experimentalmente nos ofrece buenos resultados para CIFAR sin ser excesivamente costosa de entrenar.

La red en particular consta de 10 capas convolucionales, compuestas por una convolución, una capa de normalización por lotes y una activación no lineal ReLU, seguidas por un clasificador final. Estas 10 capas se reparten en 5 bloques, cada uno de ellos delimitado por capas MaxPool que reducen la resolución del mapa de características por la mitad. El clasificador final se compone de 3 capas densamente conectadas en la estructura original, pero en nuestro caso lo reducimos a una sola, considerando la pequeña resolución de las imágenes de CIFAR, siguiendo el ejemplo de [19].

Para nuestros experimentos, consideraremos como potenciales puntos de inserción de capas OWA los puntos justo antes de cada capa convolucional, a excepción de la primera, resultando en 9 posibles puntos de inserción. La red de referencia será siempre la original sin ninguna inserción, y introduciremos capas en los puntos de inserción para generar las configuraciones de estudio. Esta estructura se refleja en la Tabla I.

Tabla I
ARQUITECTURA DE LA RED*.

Nombre	Tamaño de núcleo	Paso	Tamaño de salida
input_data	-	-	$32 \times 32 \times 3$
conv1_1	3×3	1	$32 \times 32 \times 64$
OWA_1	-	-	$32 \times 32 \times (64 + C_f)$
conv1_2	3×3	1	$32 \times 32 \times 64$
maxpool	2×2	2	$16 \times 16 \times 64$
OWA_2	-	-	$16 \times 16 \times (64 + C_f)$
conv2_1	3×3	1	$16 \times 16 \times 128$
OWA_3	-	-	$16 \times 16 \times (128 + C_f)$
conv2_2	3×3	1	$16 \times 16 \times 128$
maxpool	2×2	2	$8 \times 8 \times 128$
OWA_4	-	-	$8 \times 8 \times (128 + C_f)$
conv3_1	3×3	1	$8 \times 8 \times 256$
OWA_5	-	-	$8 \times 8 \times (256 + C_f)$
conv3_2	3×3	1	$8 \times 8 \times 256$
maxpool	2×2	2	$4 \times 4 \times 256$
OWA_6	-	-	$4 \times 4 \times (256 + C_f)$
conv4_1	3×3	1	$4 \times 4 \times 512$
OWA_7	-	-	$4 \times 4 \times (512 + C_f)$
conv4_2	3×3	1	$4 \times 4 \times 512$
maxpool	2×2	2	$2 \times 2 \times 512$
OWA_8	-	-	$2 \times 2 \times (512 + C_f)$
conv5_1	3×3	1	$2 \times 2 \times 512$
OWA_9	-	-	$2 \times 2 \times (512 + C_f)$
conv5_2	3×3	1	$2 \times 2 \times 512$
maxpool	2×2	2	$1 \times 1 \times 512$
flatten	-	-	512
linear	-	-	10

* Las capas marcadas como OWA_x son los posibles puntos de inserción para las nuevas capas OWA.

V-C. Detalles de Implementación

La implementación de estos experimentos se ha realizado en PyTorch 1.3.1 y Fastai 1.0.58.

Para todas las configuraciones empleamos los mismos hiperparámetros, en concreto, un ratio de aprendizaje máximo de $1e^{-2}$ con una política de entrenamiento 1cycle [20]. Este parámetro se ha determinado empleando la herramienta *lr_finder* de Fastai, optimizándolo para la red de referencia (sin capas OWA). Para todos los experimentos se ha empleado un tamaño de lote de 1024.

Adicionalmente hemos empleado aumentación de datos, siguiendo el ejemplo de [21]. En particular, hemos realizado volteos horizontales con una probabilidad de 0.5, y padding de 4 píxeles (empleado espejado para rellenar las regiones externas) seguido de un recorte aleatorio a la resolución original (32×32).

V-D. Evaluación

Para la evaluación de los resultados de los experimentos, optamos por repetir el entrenamiento de cada configuración 50 veces, cada una de ellas de cero (reinicializando la red) y entrenando por 30 épocas. De estos resultados recogemos la precisión en test final de cada repetición, y empleamos la media y desviación estándar de esas precisiones como medidas principales. Para comparar los resultados de las configuraciones modificadas respecto de la original empleamos el test no paramétrico de Mann-Whitney U [22]. Calculamos este test considerando como hipótesis nula que la referencia obtiene precisiones mayores o iguales que la versión modificada.

La referencia es siempre una versión sin modificar de la red. En el tercer experimento, además, entrenamos la red con dos ordenaciones de canal de referencia (no ordenarlos y ordenarlos aleatoriamente).

V-E. Configuración de los experimentos

Dada la gran cantidad de combinaciones de parámetros posibles hemos dividido el trabajo en 3 experimentos principales. Cada experimento pretende determinar uno de los principales parámetros de la capa OWA: posición, cantidad de operadores y métrica de orden. Cada experimento se repite con CIFAR10 y con CIFAR100.

1. Posición de la capa. En el primer experimento, mantenemos constante el número de operadores aprendidos ($C_f = 16$) y generamos configuraciones para todas las posiciones de capa posibles (OWA_1 a OWA_9). Como métrica de orden consideramos la suma de activaciones.
2. Número de operadores. En el segundo experimento probamos una variedad de números de operadores ($C_f = 4$,

$C_f = 8$, $C_f = 16$ y $C_f = 32$) para las dos mejores posiciones del experimento anterior, ambas con las mismas métricas.

3. Métrica de ordenación. En el tercer experimento nos centramos en las métricas de orden, fijando las mejores configuraciones de posición y número de operadores del experimento anterior. En este experimento, además, estudiamos las matrices de pesos de los operadores OWA aprendidos en la red.

VI. Estudio experimental

VI-A. *Posición de la capa*

Los resultados del primer experimento se recogen en la Tabla II. Podemos observar como existe una fuerte dependencia entre el punto de inserción y la precisión obtenida. En el caso de CIFAR10 observamos como todos los puntos de inserción entre OWA_2 y OWA_5 obtienen una mejora de precisión estadísticamente significativa respecto de la media, con el mejor en OWA_4. En CIFAR100 podemos observar resultados similares, con las mejores configuraciones en OWA_3 y OWA_5.

Sospechamos que esta tendencia a obtener mejores resultados en las capas inferiores está ligada al tamaño de imagen muy reducido de nuestro dataset, de 32×32 píxeles. Esto, en combinación con la arquitectura VGG, hace que las capas superiores tengan tamaños de imagen realmente pequeños (4×4 y menores a partir de OWA_6), haciendo que las métricas de capa no aporten información respecto de las convoluciones estándar.

Tabla II
Resultados en función del punto de inserción.

Capa	CIFAR10 prec.	CIFAR100 prec.
referencia	92.44 ± 0.17	69.74 ± 0.27
OWA_1	92.40 ± 0.19	$69.85 \pm 0.29^\bullet$
OWA_2	$92.53 \pm 0.19^\bullet$	$69.87 \pm 0.32^\bullet$
OWA_3	$92.52 \pm 0.18^\bullet$	$\mathbf{69.97 \pm 0.27}^\bullet$
OWA_4	$\mathbf{92.55 \pm 0.18}^\bullet$	$69.95 \pm 0.24^\bullet$
OWA_5	$92.51 \pm 0.17^\bullet$	$69.97 \pm 0.28^\bullet$
OWA_6	92.45 ± 0.20	69.75 ± 0.33
OWA_7	92.44 ± 0.17	69.82 ± 0.25
OWA_8	92.44 ± 0.18	69.79 ± 0.30
OWA_9	92.49 ± 0.20	69.78 ± 0.25

Los resultados señalados con $^\bullet$ mejoran la referencia con p-valor < 0.05.

VI-B. *Número de operadores*

Los resultados del segundo experimento se resumen en la Tabla III. A partir de los resultados del primer experimento, decidimos seguir explorando los puntos de inserción en OWA_3 y OWA_4 y mantenemos la métrica de ordenación de suma de activaciones.

En estos resultados observamos una cierta tendencia a favorecer mayores números de operadores aprendidos, pero sin una variación particularmente significativa. Todos los resultados para $C_f = 8$ y $C_f = 16$, en ambos datasets, obtienen resultados estadísticamente significativos con p-valor < 0.05.

Tabla III
Resultados según número de operadores.

Capa	C_f	CIFAR10 prec.	CIFAR100 prec.
referencia	-	92.44 ± 0.17	69.74 ± 0.27
OWA_3	4	92.49 ± 0.19	69.82 ± 0.32
	8	$92.51 \pm 0.21^\bullet$	$69.88 \pm 0.26^\bullet$
	16	$92.52 \pm 0.18^\bullet$	$\mathbf{69.97 \pm 0.27}^\bullet$
	32	$\mathbf{92.57 \pm 0.16}^\bullet$	69.82 ± 0.28
OWA_4	4	92.47 ± 0.17	69.81 ± 0.33
	8	$92.50 \pm 0.19^\bullet$	$69.91 \pm 0.31^\bullet$
	16	$92.55 \pm 0.18^\bullet$	$69.95 \pm 0.24^\bullet$
	32	$92.51 \pm 0.17^\bullet$	$69.90 \pm 0.29^\bullet$

Los resultados señalados con $^\bullet$ mejoran la referencia con p-valor < 0.05.

VI-C. *Métricas de orden*

Los resultados del tercer experimento se recogen en la Tabla IV. En este experimento se han empleado, en base a los resultados de los anteriores experimentos, el punto de inserción OWA_3 con $C_f = 32$ para CIFAR10 y el punto OWA_3 con $C_f = 16$ para CIFAR100. Podemos observar que la suma de activaciones obtiene mejores resultados de precisión que el resto de métricas, seguida de cerca por la variación total, tanto en CIFAR10 como en CIFAR100.

Las dos medidas de referencia, la ordenación aleatoria y la no ordenación de canales, obtienen resultados similares a la referencia sin capa OWA, de forma consistente con la hipótesis de que la capa OWA introduce nueva información a través de la función de ordenación.

Tabla IV
Resultados según la métrica de orden.

Orden	CIFAR10 prec.	CIFAR100 prec.
referencia	92.44 ± 0.17	69.74 ± 0.27
activ_sum	$\mathbf{92.57 \pm 0.16}^\bullet$	$\mathbf{69.97 \pm 0.27}^\bullet$
total_var	$92.55 \pm 0.19^\bullet$	$69.91 \pm 0.24^\bullet$
max_activ	$92.51 \pm 0.21^\bullet$	69.74 ± 0.28
median_activ	92.48 ± 0.19	69.84 ± 0.31
entropy	92.47 ± 0.16	69.80 ± 0.25
random	92.45 ± 0.17	69.76 ± 0.26
no_sorting	92.43 ± 0.19	69.79 ± 0.30

Los resultados señalados con $^\bullet$ mejoran la referencia con p-valor < 0.05.

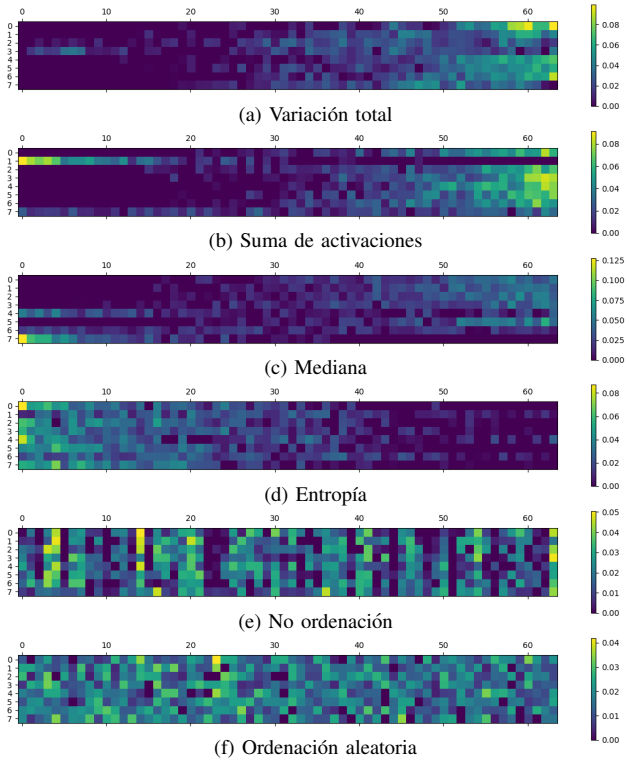
VI-D. *Matrices de pesos*

Resulta de especial interés estudiar las matrices de pesos obtenidas de los entrenamientos. En la Figura 2 mostramos 8 ejemplos de matrices de pesos de las capas OWA aprendidas, todas ellas sobre la misma configuración base (inserción en OWA_2, $C_f = 8$) y variando la métrica de orden. El tamaño de estas matrices es de 8×64, siendo $C_f = 8$ el número de operadores aprendidos (cada uno representado en una línea de la imagen) y $C_{in} = 64$ el número de pesos de cada operador, correspondientes a los canales de entrada a capa.

Podemos apreciar como el sistema converge a patrones bastante claros para la mayor parte de las agregaciones. Estos patrones se corresponden con operaciones de tipo máximo y

mínimo suavizados, donde la mayor parte del peso se reparte en las capas de entrada con mayor o menor valor de métrica asociados.

En concreto, se observa que en general el sistema converge a operaciones de tipo mínimo suavizado para todos los operadores excepto la entropía, aunque en todos los casos con algún operador de tipo máximo suavizado intercalado. En el caso de la variación total y la suma de activaciones se observan operadores con valores más concentrados, mientras que para la mediana los operadores se acercan más a una media. En el caso de la no ordenación y la ordenación aleatoria, como es de esperar, no se aprecian patrones claros.



(a) Variación total

(b) Suma de activaciones

(c) Mediana

(d) Entropía

(e) No ordenación

(f) Ordenación aleatoria

Figura 2. Operadores OWA aprendidos por la capa propuesta según diferentes métricas de orden. El eje vertical se corresponde con los diferentes operadores aprendidos, mientras que el horizontal con los pesos de los canales de entrada.

VII. Conclusiones y trabajo futuro

En este trabajo hemos propuesto la inserción de operadores OWA dentro de CNNs como un método para aumentar la información de los mapas de características. Si bien los resultados obtenidos no se posicionan en el estado del arte, consideramos que prueban sin dudas el potencial de esta técnica.

En el futuro, sería importante analizar si este enfoque puede aplicarse en redes más complejas, como ResNet [2] y arquitecturas similares, que se adaptan mejor a ciertos problemas. Es necesario, en general, una investigación más profunda sobre cómo la red neuronal está obteniendo la ventaja que hemos constatado en nuestros experimentos.

Referencias

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2012, pp. 1097–1105.

[2] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[3] R. R. Yager, "On ordered weighted averaging aggregation operators in multicriteria decisionmaking," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 18, no. 1, pp. 183–190, 1988.

[4] R. R. Yager, "Families of owa operators," *Fuzzy Sets and Systems*, vol. 59, no. 2, pp. 125 – 148, 1993.

[5] G. J. Scott, R. A. Marcum, C. H. Davis, and T. W. Nivin, "Fusion of deep convolutional neural networks for land cover classification of high-resolution imagery," *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 9, pp. 1638–1642, 2017.

[6] G. J. Scott, K. C. Hagan, R. A. Marcum, J. A. Hurt, D. T. Anderson, and C. H. Davis, "Enhanced fusion of deep neural networks for classification of benchmark high-resolution image data sets," *IEEE Geoscience and Remote Sensing Letters*, vol. 15, no. 9, pp. 1451–1455, 2018.

[7] D. T. Anderson, G. J. Scott, M. Islam, B. Murray, , and R. Marcum, "Fuzzy choquet integration of deep convolutional neural networks for remote sensing," in *Computational Intelligence for Pattern Recognition*. Springer, 2018, pp. 1–28.

[8] X. Du and A. Zare, "Multiple instance choquet integral classifier fusion and regression for remote sensing applications," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 5, pp. 2741–2753, 2019.

[9] C. A. Dias, J. C. S. Bueno, E. N. Borges, G. Lucca, H. Santos, G. P. Dimuro, H. Bustince, P. L. J. D. Junior, S. S. C. Botelho, and E. Palmeira, "Simulating the behaviour of choquet-like (pre) aggregation functions for image resizing in the pooling layer of deep learning networks," in *International Fuzzy Systems Association World Congress*. Springer, 2019, pp. 224–236.

[10] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv 1409.1556*, 2014.

[11] A. Krizhevsky, "Learning multiple layers of features from tiny images." *Master's thesis, Department ofComputer Science, University of Toronto*, 2009.

[12] J. Keller, D. Liu, and D. Fogel, *Fundamentals of Computational Intelligence: Neural Networks, Fuzzy Systems, and Evolutionary Computation*. Wiley, 2016.

[13] S. R. Price, S. R. Price, and D. T. Anderson, "Introducing fuzzy layers for deep learning," in *2019 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, 2019, pp. 1–6.

[14] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 4, pp. 640–651, 2017.

[15] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural Computation*, vol. 1, no. 4, pp. 541–551, 1989.

[16] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[17] C. E. Shannon, "A mathematical theory of communication," *The Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, 1948.

[18] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D: Nonlinear Phenomena*, vol. 60, no. 1, pp. 259 – 268, 1992.

[19] S. Liu and W. Deng, "Very deep convolutional neural network based image classification using small training sample size," in *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)*, 2015, pp. 730–734.

[20] L. N. Smith, "A disciplined approach to neural network hyper-parameters: Part 1 – learning rate, batch size, momentum, and weight decay," *arXiv 1803.09820*, 2018.

[21] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," *arXiv 1603.05027*, 2016.

[22] H. B. Mann and D. R. Whitney, "On a test of whether one of two random variables is stochastically larger than the other," *The annals of mathematical statistics*, pp. 50–60, 1947.

Learning OWA weights by combining fuzzy quantifiers with empirical data

Álvaro Cristóbal
University of Zaragoza
Zaragoza, Spain
alvcrimar@gmail.com

Ignacio Huitzil
Artificial Intelligence Research Institute (IIIA-CSIC)
Bellaterra, Spain
ihuitzil@iiia.csic.es

Fernando Bobillo
I3A, University of Zaragoza
Zaragoza, Spain
fbobillo@unizar.es

***Abstract*—Ordered Weighted Averaging (OWA) is a popular family of aggregation operators that has been used in many practical applications. This paper addresses the problem of learning the weighting vector of OWA operators and uses a hybrid approach combining a sample learning method and a function-based method. The idea is to search for the parameters of a fuzzy quantifier that minimizes the error on a given set of examples. We also perform an experimental study in the field of smart cities.**

***Index Terms*—aggregation, fuzzy quantifiers, smart cities**

I. Introduction

Aggregation of different criteria into a single value is a very common operation in many real-world applications. For example, online travel agencies typically provide a simple way for customers to compare hotels by combining the scores obtained in different criteria (such as location, price, or cleanness) into a single value. The interest in aggregation operators is not expected to decrease in the next years. For instance, in the field of smart cities, with high numbers of sensors providing pieces of information that need to be combined somehow, aggregation operators seem crucial.

Ordered Weighted Averaging (OWA) operators [1] are a very popular family of aggregation operators that has been successfully used in many applications [2]. OWA operators are parameterized with a vector of weights. While the choice of the weights is critical in the behaviour of the operators, determining the concrete values is a common problem in practice. Among the many existing solutions, we are interested in quantifier-guided aggregation [3], where the weights are computed from fuzzy quantifiers.

To illustrate the interest in quantifier-guided aggregation, we will mention some examples in the field of fuzzy ontologies, which are fuzzy extensions of the current de-facto standard for knowledge representation. For example, the fuzzy ontology language *Fuzzy OWL 2* [4] and the fuzzy ontology reasoner *fuzzyDL* [5] support quantifier-guided aggregation using right-shoulder and linear functions. Some recent applications using fuzzy ontologies also take advantage of quantifier-guided aggregation. In particular, the beer recommender system *GimmeHop* [6] and *Fuzzy BIM* [7] support flexible queries about beers and a semantic Building Information Modeling (BIM), respectively. In both cases, user preferences can be combined using OWA operators, built in a transparent way for the user from right-shoulder, linear, and power functions.

Although there have been many approaches to determine the weights of an OWA operator, the comparisons between the existing alternatives are mostly theoretical. Instead, it would be interesting to evaluate the performance of different methods on real-world datasets. In this paper, we use empirical data to learn the parameters of different functions (fuzzy quantifiers) used in quantifier-based aggregation. Therefore, our method can be seen as a combination of a sample learning method and a function-based method [8]. Moreover, we evaluate the behaviour of our learning strategy using smart cities data.

The remaining of this paper is organized as follows. Section II provides some background on aggregation operators and quantifier-based aggregation. Then, Section III describes our approach, and Section IV discusses the result of our empirical evaluation. Finally, Section V sets out some conclusions and ideas for future work.

II. Aggregation operators

Aggregation Operators (AOs) are mathematical functions that are used to combine different pieces of information (typically, membership degrees to fuzzy sets) [9], [10]. There is no standard definition of AO. Following [9], an AO takes n numerical values $x_1, x_2, \ldots, x_n$ (the values of n different criteria) and returns another numerical value, i.e., given a domain $\mathbb{D}$ (such as $[0,1]$ or $\mathbb{R}$), an AO of dimension n is a mapping $@ : \mathbb{D}^n \to \mathbb{D}$. A classical example of AO is the weighted mean. We will write $@_{\boldsymbol{W}}$ to denote the usual case where an AO is parameterized with a vector of n weights $\boldsymbol{W} = [w_1, \ldots, w_n]$ such that $w_i \in [0,1]$ and $\sum_{i=1}^{n} w_i = 1$.

A very important family of AOs are the *Ordered Weighted Averaging* (OWA) operators [1]. OWA operators provide a parameterized class of mean type AOs. Formally, given a weighting vector $\boldsymbol{W}$, an OWA operator of dimension n is an AO such that:

$$@_{\boldsymbol{W}}^{\text{owa}}(x_1, \ldots, x_n) = \sum_{i=1}^{n} w_i x_{\sigma(i)} \quad (1)$$

where σ is a permutation such that $x_{\sigma(1)} \geq x_{\sigma(2)} \geq \cdots \geq x_{\sigma(n)}$, i.e., $x_{\sigma(i)}$ is the i-th largest of the values $x_1, \ldots, x_n$

We were partially supported by the projects TIN2016-78011-C4-3-R (AEI/FEDER, UE), PID2020-113903RB-I00 (AEI/FEDER, UE), and DGA/FEDER.

to be aggregated. Note that, because of this reordering step, a weight w_i is not associated with a specific argument but with an ordered position of the aggregate. By choosing different weights, OWA operators can implement different AOs, such as arithmetic mean, k-th maximum, k-th minimum, median or order statistic, among others.

Example 1: The average temperatures of Barcelona, Basel, Logroño, Madrid, and Milan in February 2018 are 6.7, 1.4, 5.5, 6, and 7.5 ° C, respectively. Given the weighting vector $\boldsymbol{W} = [0.0048, 0.9952, 0, 0, 0]$, the aggregated value using OWA $@^{\text{owa}}_{\boldsymbol{W}}(6.7, 1.4, 5.5, 6, 7.5)$ is given by:

$$0.0048 \cdot 7.5 + 0.9952 \cdot 6.7 + 0 \cdot 6 + 0 \cdot 5.5 + 0 \cdot 1.4 = 6.70384 \blacksquare$$

A common practical problem is how to compute the weights of an OWA operator, and several solutions have been proposed in the literature [8], [11], [12]. According to X. Liu, existing approaches can be classified in 5 categories [8]: optimization-based methods, sample learning methods fitting to empirical data, function-based methods, argument dependent methods, and preference methods.

The family of function-based methods include methods to build the weights from an orness value. For example, the vector of weights $\boldsymbol{W}$ can be defined starting from a desired value for the orness in two recursive ways, a Left Recursive Form and a Right Recursive Form [13], or using Faulhaber's formulas [14]. However, the most popular example of function-based methods is *quantifier-based aggregation*.

In quantifier-based aggregation, the vector of weights $\boldsymbol{W}$ can be defined using a fuzzy quantifier $Q : [0,1] \to [0,1]$. We will focus on *Regular Increasing Monotone* (RIM) quantifiers [15], characterized by the idea that as the proportion increases, the degree of satisfaction does not decrease. More formally, RIMs satisfy the boundary conditions $Q(0) = 0$ and $Q(1) = 1$, and are monotone increasing, i.e., $x_1 \leq x_2$ implies $Q(x_1) \leq Q(x_2)$. A RIM Q can be used to define an OWA weighting vector W_Q of dimension n, where each weight is computed as follows:

$$w_i = Q(\frac{i}{n}) - Q(\frac{i-1}{n}) \tag{2}$$

Note that indeed $w_i \in [0,1]$ and $\sum_i w_i = 1$.

In this paper, we will consider the following functions to build RIMS:

- *Right-shoulder* (or *window* [16]), illustrated in Figure 1 (a). Given $q_1, q_2 \in [0,1]$ such that $q_1 < q_2$:

$$\textbf{right}(q_1, q_2) = \begin{cases} 0 & x \leq q_1 \\ \dfrac{x - q_1}{q_2 - q_1} & x \in [q_1, q_2] \\ 1 & x \geq q_2 \end{cases} \tag{3}$$

  If $q1 = q_2 \neq 1$, we have a *step* function [16]:

$$\textbf{right}(q_1, q_2) = \begin{cases} 0 & x \leq q_1 \\ 1 & x > q_1 \end{cases} \tag{4}$$

  If $q1 = q_2 = 1$, we also have a *step* function:

$$\textbf{right}(q_1, q_2) = \begin{cases} 0 & x < q_1 \\ 1 & x \geq q_1 \end{cases} \tag{5}$$

- *Linear*, illustrated in Figure 1 (b). Given $q_1, q_2 \in [0,1]$ with $q_1 \in (0,1)$:

$$\textbf{linear}(q_1, q_2) = \begin{cases} \dfrac{q_2}{q_1} \cdot x & x \leq q_1 \\ \dfrac{(1-q_2)x + (q_2 - q_1)}{1 - q_1} & x > q_1 \end{cases} \tag{6}$$

  If $q_1 = 0$:

$$\textbf{linear}(q_1, q_2) = \begin{cases} 0 & x = 0 \\ (1-q_2)x + q_2 & x > 0 \end{cases} \tag{7}$$

  If $q_1 = 1$:

$$\textbf{linear}(q_1, q_2) = \begin{cases} q_2 \cdot x & x < 1 \\ 1 & x = 1 \end{cases} \tag{8}$$

- *Power*, illustrated in Figure 1 (c). Given $q \in (0, \infty)$:

$$\textbf{power}(q) = x^q \tag{9}$$

Example 2: The weighting vector in Example 1, with $n = 5$ weights, can be computed from the quantifier $Q = \textbf{right}(0.1991, 0.3851)$:

- $w_1 = Q(1/5) - Q(0) = 0.0048 - 0 = 0.0048$
- $w_2 = Q(2/5) - Q(1/5) = 1 - 0.0048 = 0.9952$
- $w_3 = Q(3/5) - Q(2/5) = 1 - 1 = 0$
- $w_4 = Q(4/5) - Q(3/5) = 1 - 1 = 0$
- $w_5 = Q(5/5) - Q(4/5) = 1 - 1 = 0$ $\blacksquare$

III. Learning the Parameters of the Fuzzy Quantifiers

We assume that we have a set of examples $E = \langle e_1, \ldots, e_m \rangle$. Each example e_j contains the values of n input variables (x_{ij}) and the value of an output variable (y_j):

$$e_j = \langle x_{1j}, x_{2j}, \ldots, x_{nj}, y_j \rangle \tag{10}$$

For each e_j, we compute aggregated values using different weighting vectors $\boldsymbol{W}$:

$$z_j^{\boldsymbol{W}} = @^{\text{owa}}_{\boldsymbol{W}}(x_{1j}, x_{2j}, \ldots, x_{nj}) \tag{11}$$

For each $\boldsymbol{W}$, we compute the error between the expected value y_j and the aggregated value z_j. In particular, we consider the Mean Absolute Percentage Error (MAPE), which is computed as a percentage in $[0, 100]$ where the smaller the percentage, the smaller the error:

$$\text{MAPE}(E, \boldsymbol{W}) = \frac{100}{m} \sum_{j=1}^{m} \left| \frac{y_j - z_j^{\boldsymbol{W}}}{y_j} \right| \tag{12}$$

Finally, we choose the quantifier type and parameters that lead to the weighting vector $\boldsymbol{W}$ minimizing the MAPE:

$$\underset{\boldsymbol{W}}{\text{argmin}} \ \text{MAPE}(E, \boldsymbol{W}) \tag{13}$$

Example 3: Revisiting Example 1, assume that we want to predict the temperature in Zaragoza from the temperatures of the other 5 cities. If the expected value of the

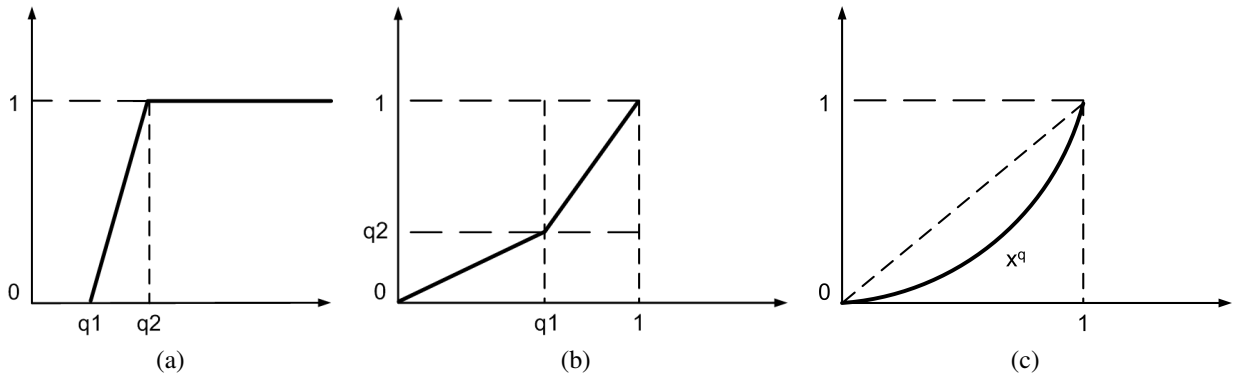


Fig. 1. (a) Right-shoulder function; (b) Linear function; (c) Power function

temperature in Zaragoza in February 2018 is 6.8 ºC, then the term of the summation corresponding to this example is $|(6.8 - 6.70384)/6.70384| = 0.0143$. ■

The key of our approach is that $\boldsymbol{W}$ is built using different types of RIM functions and with different parameters. If the search space is not very large, and computing the weights and the aggregation using OWA are not computationally expensive operations. Therefore, it could be possible to compute the best parameters by brute force, at least for not very large datasets.

- Algorithm 1 shows how to compute the parameters of a linear RIM quantifier that minimizes the MAPE using brute force. The algorithm loops over the values of q_1 and q_2 by adding an increment Δ. The best MAPE is initialized to 100, the highest possible value. When a smaller MAPE is found, the parameters q_1 and q_2 that made it possible are stored. Finally, the pair of parameters $\langle q_1, q_2 \rangle$ is returned.
- A brute-force algorithm to compute the parameters of a right-shoulder RIM quantifier is similar, but the loop over all values of q_2 starts from q_1 so that $q_1 \leq q_2$.
- Finally, for the power function, a single loop is needed, as there is just one parameter q, ranging in $(0, \infty)$.

As a final remark, note that we do not split out set of examples E into training and test sets.

For large datasets, it could be possible to use heuristic methods, such as Monte Carlo algorithms, local search, or evolutionary algorithms. For example, Algorithm 2 shows how to compute the parameters of a linear RIM quantifier using a Monte Carlo algorithm. The idea is to generate pseudo-random numbers as the possible values of q_1 and q_2, repeating the experiments several times, and storing the values that minimize the MAPE.

To conclude this section, let us note that it is trivial to consider the case where there are missing values, i.e., for some examples e_j some of the values x_{ij} are unknown. In this case, rather than having the same vector $\boldsymbol{W}$ for all the examples, we could compute for each example e_j a weighting vector of dimension n_j, where n_j is the number of non-missing values for the input variables, using the same function type and parameters for all the examples.

Algorithm 1 Brute-force algorithm to compute the parameters of a LINEAR quantifier minimizing the MAPE.

Input: A dataset E with examples as in Equation 10.
Output: Parameters $\langle q_1, q_2 \rangle$ of a linear function.

```
1:  bestMapeL ← 100
2:  for q1 ← 0 to 1 by Δ do
3:    for q2 ← 0 to 1 by Δ do
4:      Q ← linear(q1, q2)
5:      W ← compute a vector from Q using Eq. 2
6:      mape ← compute MAPE(E, W) using Eq. 12
7:      if mape < bestMapeL then
8:        bestMapeL ← mape
9:        bestQ1 ← q1
10:       bestQ2 ← q2
11:     end if
12:   end for
13: end for
14: return ⟨bestQ1, bestQ2⟩
```

IV. Evaluation

This section discusses an evaluation of our approach in the field of smart cities.

a) Datasets: We consider two datasets: Temperatures and Tourism.

- The *Temperatures* dataset includes the temperatures in 7 cities: Barcelona, Basel, Buenos Aires, Logroño, Madrid, Milan, and Zaragoza. Note that all of them are cities in Northern hemisphere except Buenos Aires, which thus has opposite seasons. Table I shows the URLs where the temperatures were retrieved. The dataset has 12 rows, with the monthly average temperatures on one year.
- The *Tourism* dataset includes information about London[1]: the total number of visits, the total number of nights, and the total spend. The values cover the period 2002–2019 (so the effects of the COVID-19 pandemic are

[1] http://data.london.gov.uk/dataset/number-international-visitors-london

TABLE I
URLS WITH THE TEMPERATURES OF EACH SMART CITY

City	URL
Barcelona	http://opendata-ajuntament.barcelona.cat/data/es/dataset/temperatures-hist-bcn
Basel	https://www.meteoblue.com/es/tiempo/archive/export/basilea_suiza_2661604
Buenos Aires	http://data.buenosaires.gob.ar/dataset/registro-temperatura-ciudad
Logroño	http://datos.gob.es/en/catalogo/a17002943-estaciones-meteorologicas-sos-rioja1
Madrid	http://es.climate-data.org/europe/espana/comunidad-de-madrid/madrid-92/t/enero-1
Milan	http://dati.comune.milano.it/dataset/ds305-ambientemeteo-temperature-mese-2008-2014
Zaragoza	http://datosclima.es/Aemet2013/Temperatura2013.php

Algorithm 2 Monte Carlo algorithm to compute the parameters of a linear quantifier minimizing the MAPE.

Input: A dataset E with examples as in Equation 10, and the number of repetitions MAX_REPETITIONS.
Output: Parameters $\langle q_1, q_2 \rangle$ of a linear function.

```
bestMapeL ← 100
repetition ← 0
repeat
    q1 ← random number in [0, 1]
    q2 ← random number in [0, 1]
    Q ← linear(q1, q2)
    W ← compute a vector from Q using Eq. 2
    mape ← compute MAPE(E, W) using Eq. 12
    if mape < bestMapeL then
        BestMapeL ← mape
        BestQ1 ← q1
        BestQ2 ← q2
    end if
    repetition ← repetition + 1
until repetition = MAX_REPETITIONS
return ⟨bestQ1, bestQ2⟩
```

not observed) and are aggregated by quarters. Therefore, there are 72 rows.

b) Experiments: We consider two experiments with the Temperatures dataset, and three with the Tourism dataset:

- E1. Prediction of the temperature in Zaragoza from the temperatures of 5 cities: Barcelona, Basel, Logroño, Madrid, and Milan.
- E2. Prediction of the temperature in Zaragoza from the temperatures of 6 cities: Barcelona, Basel, Buenos Aires, Logroño, Madrid, and Milan. This experiments is similar to E1 but taking Buenos Aires into account.
- E3. Prediction of the total number of visits from the total number of nights and the total spend.
- E4. Prediction of the total number of nights from the total number of visits and the total spend.
- E5. Prediction of the total spend form the total number of visits and the total number of nights.

In the Tourism dataset we apply a normalization step, since the three variables have a different range of values. For each variable, we divide each value by the maximum value plus a 5 %, obtaining a value in $[0, 1]$.

c) Parameters:

- In our brute-force algorithms, we use increments $\Delta = 0.001$ and $\Delta = 0.0001$.
- For the power quantifier, we take 20 as an upper bound for the value of the parameter, i.e. $q \in (0, 20]$. This choice was made after checking experimentally that higher values produce very small changes in a vector of 5 weights.
- In our Monte Carlo algorithms, we repeat the experiments $5 \cdot 10^5$ times. This choice was made to have a similar running time as in the brute-force algorithm with $\Delta = 0.001$. We also noticed that in different runs of the algorithm, the MAPE did not change if we rounded to two decimals.

d) Environmental setup: Our code was implemented in Java 1.8. All experiments were performed on a laptop computer with Intel Core i7-8750H, 16 GB RAM, 1 TB HDD + 256 GB SSD under Windows 10, 64-bits.

e) Results: Table II includes the results. For each experiment, for each algorithm, and for each quantifier type, we show the best MAPE, the best parameters, and the corresponding weighting vector. For each experiment and algorithm, we show the total running time (in seconds) to optimize the parameters of all quantifier types.

Figures 2, 3, and 4 illustrate a summary of the results for the brute-force algorithm in experiment E1, for different types of quantifiers (right-shoulder, linear, and power, respectively) and an increment $\Delta = 0.1$. MAPE values are rounded to the next integer. In Figure 4 the rows and the columns indicate the integer part and the fractional part of q, respectively, and the integer part is not shown if higher than 10.

f) Discussion: To start with, it is worth to clarify that our main objective is not to solve some prediction problems but to find the parameters corresponding to the best OWA weights. Indeed, to solve these prediction problems, more complex machine learning strategies are very likely to perform with better results.

The first interesting observation is that the MAPE is very similar regardless of the algorithm for a given quantifier type. In all the experiments with the Tourism dataset (E3, E4, and E5), the MAPE is actually the same (and so is the weighting vector) regardless of the quantifier type. In the other experiments, the differences in the MAPE are smaller than 0.007.

TABLE II
RESULTS OF THE EXPERIMENTS

Experiment	Algorithm	Quantifier	MAPE	Parameters	Weights	Time
E1	Brute-force $\Delta = 0.0001$	Right-shoulder	**5.3888**	0.1991 0.3851	[0.0048 0.9952 0 0 0]	47
		Linear	9.4604	0.4434 0.9319	[0.4203 0.4203 0.1104 0.0245 0.0245]	
		Power	10.0735	0.3014	[0.6156 0.143 0.0986 0.0777 0.065]	
	Brute-force $\Delta = 0.001$	Right-shoulder	5.3956	0.199 0.4	[0.005 0.995 0 0 0]	0.6
		Linear	9.4614	0.443 0.932	[0.4208 0.4208 0.1096 0.0244 0.0244]	
		Power	10.0741	0.301	[0.616 0.1429 0.0985 0.0776 0.065]	
	Monte Carlo	Right-shoulder	5.3893	0.1995 0.3007	[0.0048 0.9952 0 0 0]	0.6
		Linear	9.461	0.4438 0.9327	[0.4203 0.4203 0.1110 0.0242 0.0242]	
		Power	10.0736	0.3013	[0.6157 0.143 0.0986 0.0776 0.065]	
E2	Brute-force $\Delta = 0.0001$	Right-shoulder	**5.2896**	0.166 0.513	[0.0019 0.4803 0.4803 0.0375 0 0]	60
		Linear	12.6139	0.0836 0.0082	[0.0981 0.1804 0.1804 0.1804 0.1804 0.1804]	
		Power	13.4133	102.431	[0.1078 0.1474 0.1673 0.1816 0.1931 0.2028]	
	Brute-force $\Delta = 0.001$	Right-shoulder	**5.2896**	0.166 0.513	[0.0019 0.4803 0.4803 0.0375 0 0]	0.7
		Linear	12.6139	0.125 0.053	[0.0981 0.1804 0.1804 0.1804 0.1804 0.1804]	
		Power	13.4136	10.243	[0.1078 0.1474 0.1673 0.1816 0.1931 0.2028]	
	Monte Carlo	Right-shoulder	5.29	0.166 0.5129	[0.0019 0.4805 0.4805 0.0372 0 0]	0.7
		Linear	12.6139	0.1447 0.0743	[0.0981 0.1804 0.1804 0.1804 0.1804 0.1804]	
		Power	13.4133	1.2431	[0.1078 0.1474 0.1673 0.1816 0.1931 0.2028]	
E3	Brute-force $\Delta = 0.0001$	Right-shoulder	9.007	0.1492 0.5278	[0.9266 0.0734]	112
		Linear	9.007	0.0535 0.861	[0.9266 0.0734]	
		Power	9.007	0.11	[0.9266 0.0734]	
	Brute-force $\Delta = 0.001$	Right-shoulder	9.007	0.235 0.521	[0.9266 0.0734]	1.5
		Linear	9.007	0.142 0.874	[0.9266 0.0734]	
		Power	9.007	0.11	[0.9266 0.0734]	
	Monte Carlo	Right-shoulder	9.007	0.3208 0.5142	[0.9266 0.0734]	1.3
		Linear	9.007	0.4624 0.921	[0.9266 0.0734]	
		Power	9.007	0.11	[0.9266 0.0734]	
E4	Brute-force $\Delta = 0.0001$	Right-shoulder	9.8532	0.1265 0.501	[0.9973 0.0027]	105
		Linear	9.8532	0.0263 0.9948	[0.9973 0.0027]	
		Power	9.8532	0.0038	[0.9974 0.0026]	
	Brute-force $\Delta = 0.001$	Right-shoulder	9.8532	0.126 0.501	[0.9973 0.0027]	1.4
		Linear	9.8532	0.251 0.996	[0.9973 0.0027]	
		Power	9.8533	0.004	[0.9972 0.0028]	
	Monte Carlo	Right-shoulder	9.8532	0.0612 0.5012	[0.9973 0.0027]	1.4
		Linear	9.8532	0.1962 0.9957	[0.9973 0.0027]	
		Power	9.8532	0.0039	[0.9973 0.0027]	
E5	Brute-force $\Delta = 0.0001$	Right-shoulder	16.582	0.5 0.5	[0 1]	108
		Linear	16.582	0.5 0	[0 1]	
		Power	16.582	20	[0 1]	
	Brute-force $\Delta = 0.001$	Right-shoulder	16.582	0.5 0.5	[0 1]	1.3
		Linear	16.582	0.5 0	[0 1]	
		Power	16.582	20	[0 1]	
	Monte Carlo	Right-shoulder	16.582	0.6822 0.842	[0 1]	1.2
		Linear	16.582	0.5584 0	[0 1]	
		Power	16.582	20	[0 1]	

Brute-force with $\Delta = 0.0001$ always has the smaller MAPE. Brute-force with a higher increment is slightly worse in 5 cases (in E1 and in E2 for two quantifier types) and Monte Carlo is slightly worse in 4 cases (in E1 and in E2 for one quantifier type). If we compare the two worse algorithms, Monte Carlo wins strictly speaking in 4 cases and loses in 1, but the MAPEs are actually equal if we round to 2 decimals.

However, reasoning times can be more different. The fastest algorithms are brute-force with $\Delta = 0.001$ and Monte Carlo algorithm, both of them with almost the same time. On the other hand, brute-force with $\Delta = 0.0001$ is pretty much slower, and the increase in the running time is not compensated with a significant decrease in the MAPE.

If we compare the quantifier types, right-shoulder is always the best function in the Temperatures dataset, whereas in the Tourism dataset the same MAPE is always obtained regardless of the quantifier type.

Now let us give a closer look to each dataset. In the Temperatures dataset, the MAPE is in general clearly smaller in E1 than in E2. This result is expected, because E2 introduces data (from Buenos Aires) making the prediction harder. However, experiments with the right-shoulder function are an exception, and the MAPE is actually slightly smaller in E2. We can notice that in these cases the weight associated to the smallest value to be aggregated is 0, and the weight associated to the largest value is very small, about 0.002. Because Buenos Aires is the only city from the Southern hemisphere, its temperature will usually be either the highest or the lowest one, but it will have a small influence in the aggregated value because of the weights. Note that after considering Buenos Aires, the size of the weighting vector increases. Therefore, to build the weighting vectors, the quantifier functions are evaluated

q1\q2	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
0	13	13	13	11	10	10	11	12	14	16	19
0,1		13	13	10	9	10	12	13	15	18	20
0,2			13	6	6	9	11	13	15	18	21
0,3				6	6	11	13	15	17	21	24
0,4					6	17	17	19	20	25	28
0,5						17	17	20	22	27	31
0,6							17	25	25	31	36
0,7								25	25	36	42
0,8									25	66	66
0,9										66	66
1											66

Fig. 2. MAPE for right-shoulder quantifiers in E1

q1\q2	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
0	19	17	15	14	12	11	10	11	11	12	13
0,1	20	19	17	15	13	11	10	10	11	12	13
0,2	21	20	19	17	15	13	11	10	11	12	13
0,3	24	22	20	19	17	15	13	11	10	10	11
0,4	28	25	23	21	19	16	14	12	10	9	10
0,5	31	27	25	23	21	19	16	14	13	11	10
0,6	36	30	28	26	23	21	19	16	14	12	11
0,7	42	35	32	29	26	23	21	19	16	14	12
0,8	66	50	42	36	31	28	24	21	19	16	14
0,9	66	51	43	38	33	30	26	24	21	19	16
1	66	53	45	39	35	31	28	26	23	21	19

Fig. 3. MAPE for linear quantifiers in E1

in the points $\{0, 0.2, 0.4, 0.6, 0.8, 1\}$ rather than in the points $\{0, 0.25, 0.5, 0.75, 1\}$.

In the Tourism dataset, the best strategy seems using almost exclusively one of the two input variables. Indeed, in E5, the weighting vector is $[0, 1]$. In the other experiments E3 and E4, the highest value to be aggregated has a weight greater than 0.9. The MAPE is worse than in the Temperatures dataset, an is particularly high when predicting the total spend (E5).

V. Conclusions and Future Work

In this paper we have followed a hybrid approach to learn the weights of OWA operators by choosing the parameters of some functions, commonly used in quantifier-guided aggregation, that minimize the error over a set of samples. We studied right-shoulder, linear, and power functions, and proposed two alternatives for the search on the parameter space: brute force and a Monte Carlo algorithm.

q	.0	.1	.2	.3	.4	.5	.6	.7	.8	.9
0		12	11	10	11	13	14	15	16	18
1	19	19	20	21	22	22	23	23	24	24
2	25	25	26	26	27	28	28	29	30	31
3	31	32	33	33	34	34	35	35	36	36
4	37	37	38	38	38	39	39	40	40	40
5	41	41	42	42	42	43	43	43	44	44
6	44	45	45	45	46	46	46	47	47	47
7	47	48	48	48	49	49	49	49	50	50
8	50	50	51	51	51	51	52	52	52	52
9	53	53	53	53	54	54	54	54	54	55
10	55	55	55	55	56	56	56	56	56	57

Fig. 4. MAPE for power quantifiers in E1

Our approach has several advantages over learning the OWA weights directly. On the one hand, the number of parameters is smaller, reducing the search space. On the other hand, the results of more interpretable, as fuzzy quantifiers can be more easily understood by humans.

We have also discussed the results of an empirical evaluation on two datasets in the field of smart cities. We found significant differences in the running times of the algorithms but not on the error. Furthermore, we observed that in one dataset the error was smaller for right-shoulder functions, whereas in the other dataset the results were independent on the function type.

There are a lot of directions for our future work. Firstly, it would be interesting to consider more types of RIMS. Secondly, we could consider more complex algorithms to search for the best parameters. We could also take into account alternative approaches than quantifier-guided aggregation. Finally, experiments on more real-world datasets are desirable.

References

[1] R. R. Yager, "On ordered weighted averaging aggregation operators in multicriteria decision making," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 18, no. 1, pp. 183–190, 1988.

[2] R. R. Yager, J. Kacprzyk, and G. Beliakov, Eds., *Recent developments in the Ordered Weighted Averaging operators: Theory and practice*, ser. Studies in Fuzziness and Soft Computing. Springer, 2011, vol. 265.

[3] R. R. Yager, "Quantifier guided aggregation using OWA operators," *International Journal of Intelligent Systems*, vol. 11, no. 1, pp. 49–73, 1996.

[4] F. Bobillo and U. Straccia, "Fuzzy ontology representation using OWL 2," *International Journal of Approximate Reasoning*, vol. 52, no. 7, pp. 1073–1094, 2011.

[5] ——, "The fuzzy ontology reasoner fuzzyDL," *Knowledge-Based Systems*, vol. 95, pp. 12–34, 2016.

[6] I. Huitzil, F. Alegre, and F. Bobillo, "GimmeHop: A recommender system for mobile devices using ontology reasoners and fuzzy logic," *Fuzzy Sets and Systems*, vol. 401, pp. 55–77, 2020.

[7] I. Huitzil, M. Molina-Solana, J. Gómez-Romero, and F. Bobillo, "Minimalistic fuzzy ontology reasoning: An application to Building Information Modeling," *Applied Soft Computing*, vol. 103, p. 107158, 2021.

[8] X. Liu, "A review of the OWA determination methods: Classification and some extensions," in *Recent Developments in the Ordered Weighted Averaging Operators: Theory and Practice*, ser. Studies in Fuzziness and Soft Computing. Springer, 2011, vol. 265, pp. 49–90.

[9] V. Torra and Y. Narukawa, *Modeling decisions - Information fusion and aggregation operators*. Springer, 2007.

[10] G. Beliakov, H. Bustince, and T. Calvo, *A practical guide to averaging functions*, ser. Studies in Fuzziness and Soft Computing. Springer, 2016, vol. 329.

[11] Z. Xu, "An overview of methods for determining OWA weights," *International Journal of Intelligent Systems*, vol. 20, no. 8, pp. 843–865, 2005.

[12] R. Fullér, "On obtaining OWA operator weights: A sort survey of recent developments," in *Proceedings of the 5th International Conference on Computational Cybernetics (ICCC 2007)*, 2007, pp. 241–244.

[13] L. Troiano and R. R. Yager, "Recursive and iterative OWA operators," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 13, no. 6, pp. 579–600, 2005.

[14] O. Duarte and S. Téllez, "A family of OWA operators based on Faulhaber's formulas," arXiv 1801.10545, 2018.

[15] R. R. Yager, "Connectives and quantifiers in fuzzy sets," *Fuzzy Sets and Systems*, vol. 40, no. 1, pp. 39–75, 1991.

[16] ——, "Families of OWA operators," *Fuzzy Sets and Systems*, vol. 59, no. 2, pp. 125–148, 1993.

Integrales de Choquet y capacidades 2-aditivas en la construcción de indicadores sintéticos

Bonifacio Llamazares
Departamento de Economía Aplicada
Universidad de Valladolid
Valladolid
boni@eco.uva.es

Patrizia Pérez-Asurmendi
Departamento de Análisis Económico y Economía Cuantitativa
Universidad Complutense de Madrid
Madrid
patrizip@ucm.es

I. Resumen

Los indicadores sintéticos (también denominados compuestos) se han convertido en una herramienta muy valiosa en diversos ámbitos científicos dado que permiten resumir en un único valor la información proporcionada por diversos indicadores. A la hora de elaborar un indicador sintético es conveniente seguir una serie de pasos (véase [1, Tabla 1]), entre los que se encuentra la selección de un procedimiento de agregación. En la elección de dicho procedimiento resulta fundamental tener en cuenta tanto la posible correlación existente entre los indicadores utilizados como la compensabilidad que se desea permitir entre ellos (véanse [2, p. 363], [1, p. 21]).

En este contexto, uno de los métodos más habituales para agregar los valores de los indicadores consiste en utilizar funciones de agregación. Entre la gran variedad de funciones existentes destacan, por su sencillez y propiedades, las medias aritméticas y geométricas.[1] Aunque estas funciones están presentes en la construcción de muchos indicadores sintéticos, su uso ha sido objeto de diversas críticas.

En el caso de la media aritmética, la principal tiene que ver con el hecho de que valores elevados en algunos indicadores pueden compensar valores bajos de otros. Ello ha originado que, en algunos casos, haya sido sustituida por la media geométrica. Por ejemplo, el Índice de Desarrollo Humano (IDH), que hasta el año 2009 se basaba en la media aritmética, se construye desde el año 2010 mediante la media geométrica debido a que esta función permite una menor compensabilidad entre los indicadores que la media aritmética. Por lo que respecta a la media geométrica, sus principales debilidades son la imposibilidad de utilizar el 0 como valor mínimo de referencia (algo bastante habitual en la práctica al transformar los valores de los indicadores mediante determinadas normalizaciones), lo cual dificulta la capacidad informativa de la escala utilizada, y que no es invariante a cambios de origen [3].

En el contexto descrito, la integral de Choquet [4] emerge como un instrumento eficaz para solventar los problemas anteriormente mencionados. Además, permite tener en cuenta tanto la posible correlación existente entre los indicadores utilizados como la compensabilidad que se desea permitir entre ellos. En la definición de la integral de Choquet juega un papel fundamental el concepto de capacidad. Entre la amplia variedad de capacidades que se pueden construir, las capacidades 2-aditivas [5] son, probablemente, el mejor compromiso entre baja complejidad y riqueza del modelo [6].

En trabajos anteriores (véanse [7], [8]) se ha desarrollado un modelo basado en capacidades 2-aditivas que permite tener en cuenta las interacciones que habitualmente existen entre los indicadores empleados. En el presente trabajo se extiende el modelo anterior de manera que, además de tener en cuenta las interacciones existentes entre los indicadores, es posible regular la compensabilidad que se permite entre ellos.

Como aplicación práctica, el modelo propuesto se aplica al Índice de Sociedad Sostenible (SSI),[2] reemplazando, como función de agregación, la media geométrica por la integral de Choquet y comparando los resultados obtenidos según ambas agregaciones.

Bibliografía

[1] M. Nardo, M. Saisana, A. Saltelli, S. Tarantola, A. Hoffmann, and E. Giovannini, *Handbook on Constructing Composite Indicators: Methodology and User Guide*. OECD Publishing, 2008.

[2] A. Saltelli, M. Nardo, M. Saisana, and S. Tarantola, "Composite indicators — The controversy and the way forward," in *Statistics, Knowledge and Policy: Key Indicators to Inform Decision Making (Proceedings of the World Forum on Key Indicators)*. Palermo (Italy): OECD, November 10-13, 2004 2005, pp. 359–372.

[3] M. Grabisch, "The application of fuzzy integrals in multicriteria decision making," *European Journal of Operational Research*, vol. 89, no. 3, pp. 445–456, 1996.

[4] G. Choquet, "Theory of capacities," *Annales de l'Institut Fourier*, vol. 5, pp. 131–295, 1953.

[5] M. Grabisch, "k-order additive discrete fuzzy measures and their representation," *Fuzzy Sets and Systems*, vol. 92, no. 2, pp. 167–189, 1997.

[6] M. Grabisch and C. Labreuche, "A decade of application of the Choquet and Sugeno integrals in multi-criteria decision aid," *Annals of Operations Research*, vol. 175, no. 1, pp. 247–286, 2010.

[7] B. Llamazares and P. Pérez-Asurmendi, "Criteria interaction and Shapley values," in *XIII Encuentro de la Red Española de Elección Social (REES)*, Granada, 18-19 de noviembre de 2016.

[8] B. Llamazares and P. Pérez-Asurmendi, "Elaboración de indicadores sintéticos mediante la integral de Choquet," in *XXV Jornadas de ASEPUMA*, La Coruña, 8-9 de junio de 2017.

Patrizia Pérez-Asurmendi agradece el apoyo recibido por la Fundación Areces (Proyecto CISP18A6216).

[1] En ambas funciones, la importancia de los indicadores en el indicador sintético se establece con el vector de pesos utilizado en su cálculo.

[2] Este indicador sintético permite medir la sostenibilidad en un sentido amplio dado que tiene en cuenta tres dimensiones del bienestar: el bienestar humano, el bienestar medioambiental y el bienestar económico.

Extensión multidimensional de la integral de Choquet discreta y su aplicación en redes neuronales recurrentes

Mikel Ferrero-Jaurrieta, Iosu Rodríguez-Martínez, Graçaliz Pereira Dimuro, Javier Fernández, Humberto Bustince
Departamento de Estadística, Informática y Matemáticas
Universidad Pública de Navarra
Campus Arrosadía, s/n, 31006 Pamplona, España
{mikel.ferrero, iosu.rodriguez, gracaliz.pereira, fcojavier.fernandez, bustince}@unavarra.es

***Abstract*—En este trabajo presentamos una definición de la integral de Choquet discreta n-dimensional, para fusionar datos vectoriales. Como aplicación, utilizamos estas nuevas integrales de Choquet discretas multidimensionales en la fusión de información secuencial en las redes neuronales recurrentes, mejorando los resultados obtenidos mediante el método de agregación tradicional.**

***Index Terms*—Integral de Choquet, Función de Agregación, Información Multivariante, Redes Neuronales Recurrentes, LSTM.**

I. Introducción

El proceso de fusión de información resulta un procedimiento fundamental a la hora de combinar o agregar distintas estructuras de información en una sola [1]. Su utilización es necesaria en diversos campos, como por ejemplo: toma de decisión multi-criterio [2], economía y finanzas [3], estadística, procesamiento de imágenes [4], aprendizaje automático [5], etc. Recientemente también se ha aplicado en el aprendizaje profundo, por ejemplo en las capas de *pooling* de redes neuronales convolucionales [6].

En la literatura se analizan multitud de métodos de agregación de información. Algunos de los más habituales son las medias aritméticas ponderadas [7] o los órdenes estadísticos [8]. Sin embargo, frecuentemente los criterios y datos considerados interactúan entre ellos y es conveniente utilizar operadores de agregación que tengan en cuenta este hecho. En este sentido, en la literatura se han utilizado las integrales difusas [9], las cuales se basan en medidas difusas. Estas medidas [10] permiten tener en cuenta la relación existente entre los elementos a agregar, valorando la relevancia de posibles coaliciones entre los datos [5].

Una de las integrales difusas más utilizadas es la integral de Choquet [11]. Hasta ahora, en la literatura se han presentado distintas generalizaciones de la integral de Choquet [12], [13], [5], [14] para datos unidimensionales.

Frecuentemente los elementos a agregar son datos o instancias con varias variables, es decir, información multivariante, estructurada en forma de vectores. Por ejemplo, un modelo utilizado actualmente en la inteligencia artificial donde se manejan datos multivariantes son las Redes Neuronales Recurrentes [15].

Las Redes Neuronales Recurrentes son un tipo de red neuronal artificial que se encargan de modelar información de tipo secuencial o temporal, como las series temporales o el procesamiento de lenguaje natural [16]. Dichas redes constan de una arquitectura en la que en cada instante, los valores de salida de la capa del instante anterior se conectan con la información del instante actual. Para la conexión de dichos datos multivariantes, usualmente se suelen sumar los vectores como forma de agregación de la informacion multivariante secuencial. Entre los datos recurrentes generados por la red y los datos provinientes del dataset puede haber interacción entre los mismos.

El objetivo de este trabajo es presentar una extensión multidimensional de la integral de Choquet, es decir, una función que agregue m datos n-dimensionales, teniendo en cuenta las posibles coaliciones entre los mismos.

Para mostrar la utilidad de nuestra extensión, presentamos su uso en la modelización de las posibles coaliciones entre datos en el proceso de agregación de una red neuronal recurrente. En concreto, la modificación de la arquitectura que presentamos en este trabajo consiste en una red neuronal recurrente del tipo memoria de corto y largo plazo (LSTM) [17]. En los pasos en los que se agrega la información recurrente con la información inicial estudiamos la utilización de la integral de Choquet discreta multidimensional. De forma análoga, completamos el estudio con el uso de combinaciones lineales de otras funciones de agregación.

Este trabajo se organiza de la siguiente manera: en la Sección II recordamos los conceptos preliminares necesarios para comprender el resto del trabajo. En la Sección III se introduce la nueva definición de la integral de Choquet extendida a datos multivariantes. En la Sección IV se introduce la modificación en la arquitectura. En la Sección V se presenta la experimentación realizada. Por último, en la Sección VI, se explican las conclusiones así como las líneas futuras del trabajo.

II. Preliminares

En esta sección, recordamos nociones básicas y terminología necesarias para abordar el desarrollo del trabajo. Presentamos por un lado las definiciones teóricas básicas de función de agregación, medida difusa e integral de Choquet, y por otro la explicación del funcionamiento de una red neuronal recurrente de tipo LSTM.

A. Funciones de agregación e integral de Choquet

Consideremos el retículo $(L, \leq)$ donde $L = [0,1]$ y sea $\leq$ el orden natural de los números reales. Denotamos $\mathbf{0} = (0, \ldots, 0) \in L^n$ y $\mathbf{1} = (1, \ldots, 1) \in L^n$.

Definición II.1. *Sea m un entero positivo. Una función $M : L^m \to L$ es una función de agregación m-aria si satisface las siguientes propiedades:*

(i) $M(\mathbf{0}) = 0$ *y* $M(\mathbf{1}) = 1$

(ii) es no-decreciente en cada variable, es decir, para todo $(x_1, \ldots, x_m), (y_1, \ldots, y_m) \in L^m$, $M(x_1, \ldots, x_m) \leq M(y_1, \ldots, y_m)$ *si* $x_1 \leq y_1, \ldots, x_m \leq y_m$.

Denotamos el conjunto $\{1, \ldots, m\}$ por $[m]$. Dos vectores $(x_1, \ldots, x_m), (y_1, \ldots, y_m) \in L^m$ son considerados comonótonos si y sólo si existe una permutación $\sigma : [m] \to [m]$ tal que $x_{\sigma(1)} \leq \ldots \leq x_{\sigma(m)}$ e $y_{\sigma(1)} \leq \cdots \leq y_{\sigma(m)}$.

Denotamos con letras negrita los elementos en L^n, esto es, $\mathbf{x} = (x_1, \ldots, x_n) \in L^n$. Existe un orden parcial $\leq_P$ inducido por $\leq$ y dado de la siguiente manera:

$$\mathbf{x} \leq_P \mathbf{y} \text{ si y sólo si } x_i \leq y_i$$

para todo $i \in \{1, \ldots, n\}$.

De hecho, podemos verificar que $(L^n, \leq)$ es un retículo donde el elemento mínimo es $\mathbf{0}$ y el máximo es $\mathbf{1}$. En este retículo el ínfimo y el supremo de dos elementos vienen dados por las siguientes operaciones:

$$\mathbf{x} \wedge \mathbf{y} = (\min(x_1, y_1), \ldots, \min(x_n, y_n)) \tag{1}$$

$$\mathbf{x} \vee \mathbf{y} = (\max(x_1, y_1), \ldots, \max(x_n, y_n)) \tag{2}$$

Con anterioridad a la definición del concepto de integral de Choquet, consideramos la definición de medida difusa.

Definición II.2. *[18] Una medida difusa definida sobre $[m]$ es una función $\nu : 2^{[m]} \to L$ tal que:*

(i) $\nu(\emptyset) = 0$ *y* $\nu([m]) = 1$

(ii) $\nu(\mathcal{A}) \leq \nu(\mathcal{B})$ *para todo* $\mathcal{A} \subseteq \mathcal{B} \subseteq [m]$

Una medida difusa $\nu : 2^{[m]} \to L$ se dice que es aditiva si $\nu(\mathcal{A} \cup \mathcal{B}) = \nu(\mathcal{A}) + \nu(\mathcal{B})$ para todo $\mathcal{A}, \mathcal{B} \subseteq [m]$ tales que $\mathcal{A} \cap \mathcal{B} = \emptyset$.

Ejemplo II.3. *Un ejemplo de medida difusa considerada en este trabajo es la medida de potencia. Es definida para todo $\mathcal{A} \subseteq [m]$ por:*

$$\nu_q(\mathcal{A}) = \left(\frac{|\mathcal{A}|}{m}\right)^q \tag{3}$$

donde $q > 0$.

Ejemplo II.4. *La medida difusa $\nu_l : 2^{[m]} \to L$ más pequeña viene dada por*

$$\nu_l(\mathcal{A}) = \begin{cases} 1 & \text{si } \mathcal{A} = [m] \\ 0 & \text{en otro caso} \end{cases} \tag{4}$$

La medida difusa $\nu_u : 2^{[m]} \to L$ más grande viene dada por

$$\nu_u(\mathcal{A}) = \begin{cases} 0 & \text{si } \mathcal{A} = \emptyset \\ 1 & \text{en otro caso} \end{cases} \tag{5}$$

Para cualquier medida difusa $\nu : 2^{[m]} \to L$ se cumple:

$$\nu_l(\mathcal{A}) \leq \nu(\mathcal{A}) \leq \nu_u(\mathcal{A})$$

para todo $A \subseteq [m]$

Una vez introducida la medida difusa, presentamos la definición de la integral de Choquet discreta, la cual es un ejemplo de función de agregación presentada en la Definición II.1.

Definición II.5. *[11] La integral de Choquet discreta en L con respecto a la medida difusa ν es definida como una aplicación $Ch_\nu : L^m \to L$*

$$Ch_\nu(\mathbf{x}) = \sum_{i=1}^{m} \left(x_{\sigma(i)} - x_{\sigma(i-1)}\right) \nu\left(\mathcal{A}_{\sigma(i)}\right) \tag{6}$$

donde $\mathbf{x} = (x_1, \ldots, x_m) \in L^m$, $\nu : 2^{[m]} \to L$ es una medida difusa en el conjunto $[m]$, $\sigma : [m] \to [m]$ es una permutación, con $x_{\sigma(1)} \leq \ldots \leq x_{\sigma(m)}$ con la convención $x_{\sigma(0)} = 0$ y $A_{\sigma(i)} := \{\sigma(i), \ldots, \sigma(m)\}$ es el subconjunto de los índices correspondiente a los $m - i + 1$ mayores elementos de $\mathbf{x}$ para todo $i \in [m]$.

Por último, si bien definimos la integral de Choquet $Ch_\nu : L^m \to L$, en la literatura también podemos encontrar la definición [19] para un intervalo $\mathbb{I} = [a, b] \subset \mathbb{R}$. Dado que posteriormente en la aplicación (Sección IV) la utilizaremos, se puede definir la integral de Choquet discreta como una aplicación $Ch_\nu : \mathbb{I}^m \to \mathbb{I}$, mediante la misma expresión que en la Eq. 6. Donde $\mathbf{x} = (x_1, \ldots, x_m) \in \mathbb{I}^m$ con la medida difusa $\nu : 2^{[m]} \to L$, manteniendo la convención $x_{\sigma(0)} = 0$ y $\mathcal{A}_{\sigma(i)} := \{\sigma(i), \ldots, \sigma(m)\}$.

A continuación introducimos la definición de funciones de agregación n-dimensionales, como marco introductorio a la Sección III.

Definición II.6. *[20] Sean n, m enteros positivos. Una aplicación $M : (L^n)^m \to L^n$ es una función de agregación m-aria n-dimensional si satisface las siguientes propiedades:*

(i) $M(\mathbf{0}, \ldots, \mathbf{0}) = \mathbf{0}$ *y* $M(\mathbf{1}, \ldots, \mathbf{1}) = \mathbf{1}$

(ii) Para todo $\mathbf{x_1}, \ldots, \mathbf{x_m}, \mathbf{y_1}, \ldots, \mathbf{y_m} \in L^n$ *tal que* $\mathbf{x_1} \leq \mathbf{y_1}, \ldots, \mathbf{x_m} \leq \mathbf{y_m}$, *entonces* $M(\mathbf{x_1}, \ldots, \mathbf{x_m}) \leq M(\mathbf{y_1}, \ldots, \mathbf{y_m})$.

B. Redes Neuronales Recurrentes: Memoria a corto y largo plazo (LSTM)

Las Redes Neuronales Recurrentes (RNN) nacen con la intención de modelar datos que tienen dependencia secuencial o de tiempo. Sin embargo, dado que los algoritmos de aprendizaje para redes neuronales suelen están basados en el gradiente, en estas redes surge el problema llamado *vanishing gradient* [21], es decir, el decrecimiento recurrente del valor de una variable en la salida de la red neuronal. Esto es un problema especialmente grave cuando tratamos de entrenar redes con dependencias o secuencias temporales largas.

Las *Long Shot-Term Memory* (LSTM) surgen principalmente como respuesta a este problema y suponen un cambio radical [21] en el entrenamiento de las redes recurrentes ya que evitan el decrecimiento continuo de los parámetros. De esta manera, esta arquitectura de neurona artificial [17] genera un estado que permite la memorización de conocimiento que se utiliza en instantes temporales posteriores.

Las neuronas LSTM han tenido diversas modificaciones en la literatura, pero en este trabajo utilizamos una de las más extendidas [16]. En la Fig. 1 podemos observar el detalle del interior de una LSTM donde es importante recalcar las puertas [22] *forget gate* (f), *input gate* (i) y *output gate* (o) así como la celda candidata ($\tilde{c}$).

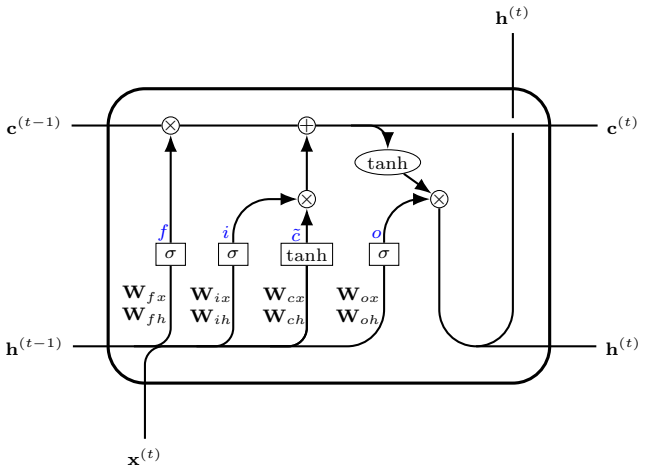


Fig. 1. Representación de una unidad LSTM.

A continuación, explicamos el funcionamiento de una unidad LSTM. Sea N la longitud de la secuencia de entrada, H el número de características que extraiga la celda y T el número de instantes de tiempo de la secuencia. Los que se detallan a continuación son las matrices y vectores asociados a cada una de las puertas y celda candidata:

- Matrices de pesos de entrada: $\mathbf{W}_{fx}, \mathbf{W}_{ix}, \mathbf{W}_{cx}, \mathbf{W}_{ox} \in \mathbb{R}^{H \times N}$
- Matrices de pesos recurrentes: $\mathbf{W}_{fh}, \mathbf{W}_{ih}, \mathbf{W}_{ch}, \mathbf{W}_{oh} \in \mathbb{R}^{H \times H}$
- Vectores de pesos del sesgo: $\mathbf{b}_f, \mathbf{b}_i, \mathbf{b}_c, \mathbf{b}_o \in \mathbb{R}^H$

El funcionamiento de forma descriptiva para cada instante de tiempo $t \in \{1, \ldots, T\}$ es el siguiente:

i. Los valores de entrada $\mathbf{x}^{(t)}, \mathbf{h}^{(t-1)}$ entran a las puertas f (Eq. 7), i (Eq. 8), $\tilde{c}$ (Eq. 9) y o (Eq. 11). En cada una de ellas, el valor $\mathbf{x}^{(t)}$ se multiplica por cada una de las matrices de pesos de entrada, en función de la puerta. De forma análoga ocurre con los valores $\mathbf{h}^{(t-1)}$ y las matrices de pesos recurrentes. Los vectores H-dimensionales obtenidos se suman junto con los vectores de sesgo correspondientes en cada una de ellas. Se utiliza una sigmoidea logística ($\sigma(x) = \frac{1}{1+e^{-x}}$) como función de activación de las puertas y la tangente hiperbólica como función de activación de la celda candidata.

ii. El vector de la memoria a largo plazo del instante anterior ($\mathbf{c}^{(t-1)}$) y el de la celda candidata ($\tilde{\mathbf{c}}^{(t)}$) se combinan. Para ello se calcula el producto de Hadamard o elemento a elemento ($\circ$) entre el valor de la *forget gate* y la *input gate* respectivamente (Eq. 10). Ambos valores se suman obteniendo el valor de la celda en el instante actual ($\mathbf{c}^{(t)}$).

iii. Por último, se calcula el vector de la memoria a corto plazo. Para ello, en primer lugar se pasa la información a corto plazo por una función de activación de salida. Para ello, utilizamos la tangente hiperbólica. Posteriormente se calcula el producto de Hadamard entre el valor de la *output gate* con la información obtenida de la última función de activación, obteniendo el valor de la memoria a corto plazo, $\mathbf{h}^{(t)}$.

Las ecuaciones que describen el proceso son las siguientes (Eq. 7-12):

$$\mathbf{f}^{(t)} = \sigma(\mathbf{W}_{fx}\mathbf{x}^{(t)} + \mathbf{W}_{fh}\mathbf{h}^{(t-1)} + \mathbf{b}_f) \tag{7}$$

$$\mathbf{i}^{(t)} = \sigma(\mathbf{W}_{ix}\mathbf{x}^{(t)} + \mathbf{W}_{ih}\mathbf{h}^{(t-1)} + \mathbf{b}_i) \tag{8}$$

$$\tilde{\mathbf{c}}^{(t)} = \tanh(\mathbf{W}_{cx}\mathbf{x}^{(t)} + \mathbf{W}_{ch}\mathbf{h}^{(t-1)} + \mathbf{b}_c) \tag{9}$$

$$\mathbf{c}^{(t)} = \mathbf{f}^{(t)} \circ \mathbf{c}^{(t-1)} + \mathbf{i}^{(t)} \circ \tilde{\mathbf{c}}^{(t)} \tag{10}$$

$$\mathbf{o}^{(t)} = \sigma(\mathbf{W}_{ox}\mathbf{x}^{(t)} + \mathbf{W}_{oh}\mathbf{h}^{(t-1)} + \mathbf{b}_o) \tag{11}$$

$$\mathbf{h}^{(t)} = \mathbf{o}^{(t)} \circ \tanh(\mathbf{c}^{(t)}) \tag{12}$$

III. Integral de Choquet multidimensional

A continuación se presenta una definición de la Integral de Choquet en retículos $(L^n, \leq)$, la cual es también un ejemplo de función de agregación n-dimensional, presentada en la Definición II.6.

Sean $\mathbf{x_1} = (x_{11}, \ldots, x_{1n}), \ldots, \mathbf{x_m} = (x_{m1}, \ldots, x_{mn})$ m vectores en L^n. Denotaremos por $\mathbf{x^1}, \ldots, \mathbf{x^n}$ a los siguientes n vectores en L^m: $\mathbf{x^1} = (x_{11}, \ldots, x_{m1}), \ldots, \mathbf{x^n} = (x_{1n}, \ldots, x_{mn})$. Es decir, por ejemplo si $\mathbf{x_1}, \ldots, \mathbf{x_m}$ son filas de una matriz de tipo $m \times n$, entonces $\mathbf{x^1}, \ldots, \mathbf{x^n}$ son las columnas de dicha matriz.

Definición III.1. *Sean n y m dos números enteros positivos. Sea $\boldsymbol{\nu} = (\nu_1, \ldots, \nu_n)$ una secuencia de medidas difusas en el conjunto $[m]$ y sean $Ch_{\nu_i} : L^m \to L$ las integrales de Choquet en L con respecto a la medida ν_i para todo $i \in [m]$. Una función $\mathbf{Ch}^r_{\boldsymbol{\nu}} : (L^n)^m \to L^n$ dada por:*

$$\mathbf{Ch}^r_{\boldsymbol{\nu}}(\mathbf{x_1}, \ldots, \mathbf{x_m}) = \big(Ch_{\nu_1}(\mathbf{x^1}), \ldots, Ch_{\nu_n}(\mathbf{x^n})\big) \quad (13)$$

para todo $\mathbf{x_1}, \ldots, \mathbf{x_m} \in L^n$ es una Integral de Choquet discreta representable en L^n con respecto a $\boldsymbol{\nu}$ y al orden $\leq$.

La integral de Choquet $\mathbf{Ch}^r_{\boldsymbol{\nu}}$ se denomina representable ya que es obtenida a través de la utilización de n integrales de Choquet en L de forma separada para cada componente:

$$\begin{aligned}\mathbf{Ch}^r_{\boldsymbol{\nu}}(\mathbf{x_1}, \ldots, \mathbf{x_m}) = (&Ch_{\nu_1}(x_{11}, \ldots, x_{m1}), \ldots, \\ &Ch_{\nu_n}(x_{1n}, \ldots, x_{mn}))\end{aligned} \quad (14)$$

Esta expresión es una generalización de la integral de Choquet estándar en L, ya que si todos los vectores de entrada son n-tuplas con las mismas coordenadas, p.e. $\mathbf{x} = (x, \ldots, x)$ y la secuencia de medidas difusas es un vector con las mismas medidas, esto es, $\nu_1 = \ldots = \nu_n = \nu$, la salida es una n-tupla con las mismas coordenadas iguales a Ch_ν.
Así mismo, de forma análoga a la que se explica en la Sección II, se puede extender la Definición III.1 a un intervalo $\mathbb{I} = [a, b] \subset \mathbb{R}$. De esta manera, podemos extender la función $\mathbf{Ch}^r_{\boldsymbol{\nu}} : (L^n)^m \to L^n$ a una aplicación $\mathbf{Ch}^r_{\boldsymbol{\nu}} : (\mathbb{I}^n)^m \to \mathbb{I}^n$ dada por:

$$\mathbf{Ch}^r_{\boldsymbol{\nu}}(\mathbf{x_1}, \ldots, \mathbf{x_m}) = \big(Ch_{\nu_1}(\mathbf{x^1}), \ldots, Ch_{\nu_n}(\mathbf{x^n})\big) \quad (15)$$

para todo $\mathbf{x_1}, \ldots, \mathbf{x_m} \in \mathbb{I}^n$ con respecto a $\boldsymbol{\nu}$ y al orden $\leq$.

Proposición III.2. *Bajo las condiciones de la definición anterior, sean $\nu_1 = \ldots = \nu_n = \nu$. Entonces:*

$$\begin{aligned}\mathbf{Ch}^r_{\boldsymbol{\nu}}(\mathbf{x_1}, \ldots, \mathbf{x_m}) = (&Ch_{\nu}(x_1, \ldots, x_m), \ldots, \\ &Ch_{\nu}(x_1, \ldots, x_m))\end{aligned} \quad (16)$$

para todo $\mathbf{x_i} = (x_i, \ldots, x_i) \in L^n$, $i \in [m]$.

IV. Modificación de la arquitectura de una red neuronal recurrente basada en la definición multidimensional de la integral de Choquet

En la presente sección se explica la introducción de las definiciones expuestas en las anteriores secciones en la arquitectura de una red neuronal recurrente, concretamente en una LSTM.
En este sentido, modificamos el operador de agregación de la red LSTM (suma de vectores) por la nueva definición de integral de Choquet discreta multidimensional.
En las nuevas ecuaciones generadas por la modificación de la función de agregación, esto se ve reflejado en el cálculo de la salida de la *forget gate* (Eq. 17), *input gate* (Eq. 18), *output gate* (Eq. 21) y celda candidata (Eq. 19), donde los vectores son fusionados mediante la integral de Choquet multidimensional.
El conjunto de las ecuaciones modificadas que describen el proceso son las siguientes:

$$\mathbf{f}^{(t)} = \sigma\left(\mathbf{Ch}^r_{\boldsymbol{\nu}}(\mathbf{W}_{fx}\mathbf{x}^{(t)}, \mathbf{W}_{fh}\mathbf{h}^{(t-1)}, \mathbf{b}_f)\right) \quad (17)$$

$$\mathbf{i}^{(t)} = \sigma\left(\mathbf{Ch}^r_{\boldsymbol{\nu}}(\mathbf{W}_{ix}\mathbf{x}^{(t)}, \mathbf{W}_{ih}\mathbf{h}^{(t-1)}, \mathbf{b}_i)\right) \quad (18)$$

$$\tilde{\mathbf{c}}^{(t)} = \tanh\left(\mathbf{Ch}^r_{\boldsymbol{\nu}}(\mathbf{W}_{cx}\mathbf{x}^{(t)}, \mathbf{W}_{ch}\mathbf{h}^{(t-1)}, \mathbf{b}_c)\right) \quad (19)$$

$$\mathbf{c}^{(t)} = \mathbf{f}^{(t)} \circ \mathbf{c}^{(t-1)} + \mathbf{i}^{(t)} \circ \tilde{\mathbf{c}}^{(t)} \quad (20)$$

$$\mathbf{o}^{(t)} = \sigma\left(\mathbf{Ch}^r_{\boldsymbol{\nu}}(\mathbf{W}_{ox}\mathbf{x}^{(t)}, \mathbf{W}_{oh}\mathbf{h}^{(t-1)}, \mathbf{b}_o)\right) \quad (21)$$

$$\mathbf{h}^{(t)} = \mathbf{o}^{(t)} \circ \tanh(\mathbf{c}^{(t)}) \quad (22)$$

Como hemos mostrado, la modificación principal del funcionamiento de la celda LSTM será la sustitución de la suma por la integral de Choquet multidimensional. No obstante, por completitud en el estudio, en la fusión de información vectorial en la unidad LSTM utilizaremos distintas funciones de agregación, así como las combinaciones lineales entre sí. Tomamos $\mathbf{z} = (\mathbf{W}_{gx}\mathbf{x}^{(t)}, \mathbf{W}_{gh}\mathbf{h}^{(t-1)}, \mathbf{b}_g) = (z_i)_{i=1}^3$ para $g \in \{f, i, c, o\}$, dado que son la misma expresión con matrices de pesos diferentes. A partir de ello, en la Tabla I mostramos las distintas funciones que utilizaremos.

TABLE I
Distintas funciones de agregación utilizadas

M	*Expresión función agregación*
Max	$\max_{i=1}^3 z_i$
Ch$_2$	$\mathbf{Ch}^r_{\boldsymbol{\nu_2}}(\mathbf{z})$
Ch$_q$	$\mathbf{Ch}^r_{\boldsymbol{\nu_q}}(\mathbf{z})$
Sum	$\sum_{i=1}^3 z_i$
Max + *Sum*	$\lambda_1 \max_{i=1}^3 z_i + \lambda_2 \sum_{i=1}^3 z_i$
Ch$_2$ + *Sum*	$\lambda_1 \mathbf{Ch}^r_{\boldsymbol{\nu_2}}(\mathbf{z}) + \lambda_2 \sum_{i=1}^3 z_i$
Ch$_q$ + *Sum*	$\lambda_1 \mathbf{Ch}^r_{\boldsymbol{\nu_q}}(\mathbf{z}) + \lambda_2 \sum_{i=1}^3 z_i$
Ch$_2$ + *Max*	$\lambda_1 \mathbf{Ch}^r_{\boldsymbol{\nu_2}}(\mathbf{z}) + \lambda_2 \max_{i=1}^3 z_i$
Ch$_q$ + *Max*	$\lambda_1 \mathbf{Ch}^r_{\boldsymbol{\nu_q}}(\mathbf{z}) + \lambda_2 \max_{i=1}^3 z_i$

Siendo $\nu_q(\mathcal{A})$ la expresión de la Eq. 3 donde q es un parámetro aprendido por la red neuronal. La medida $\nu_2(\mathcal{A})$ se corresponde a la misma expresión de la Eq. 3 evaluando $q = 2$. Los parámetros λ_1 y λ_2 son aprendidos por la propia red neuronal recurrente mediante el método de descenso por gradiente estocástico.

V. Estudio experimental

En la presente sección explicamos la arquitectura, conjunto de datos e hiperparámetos utilizados así como los resultados obtenidos y su posterior valoración.

A. Marco de trabajo experimental

1) Conjunto de datos: El conjunto de datos utilizado es el *Fashion-MNIST* [23], el cual consiste en un conjunto de entrenamiento de 60.000 imágenes de dimensiones 28×28 distribuidas en 10 clases, junto con un conjunto de test compuesto por 10.000 imágenes similares. Las imágenes corresponden a 10 artículos diferentes de categorías de ropa. En este caso, planteamos los datos que contiene una imagen como información secuencial [24].

2) Arquitectura: En este caso, como podemos observar en Fig. 2, establecemos una arquitectura en la que las imágenes son tomadas como datos secuenciales. En cada instante de tiempo $t \in \{1, \ldots, T\}$ es tomada una fila de la imagen en forma de vector como dato de entrada $\mathbf{x}^{(t)} \in [0, 255]^N$. En el caso de este dataset concreto, $T = N = 28$.
La arquitectura consta de dos capas. La primera, una unidad de memoria LSTM con $H = 128$ nodos de capa oculta. En segundo lugar, una capa totalmente conectada (FC) que conecta los 128 nodos de la unidad LSTM con 10 nodos de FC, asignándoles un valor de probabilidad en $[0, 1]$ a cada uno de ellos. Se clasifica en el número de clase correspondiente al máximo valor de probabilidad del vector softmax. Esto es, tomando $\hat{y} = arg\max \frac{exp(FC_j)}{\sum_{k=0}^{9} exp(FC_k)}$.
En este experimento se han realizado 10 ejecuciones independientes de 40 epochs cada una. La tasa de aprendizaje fijada para el experimento es de $\alpha = 0.1$ y el método de optimización utilizado para el aprendizaje ha sido el descenso por gradiente estocástico (SGD).

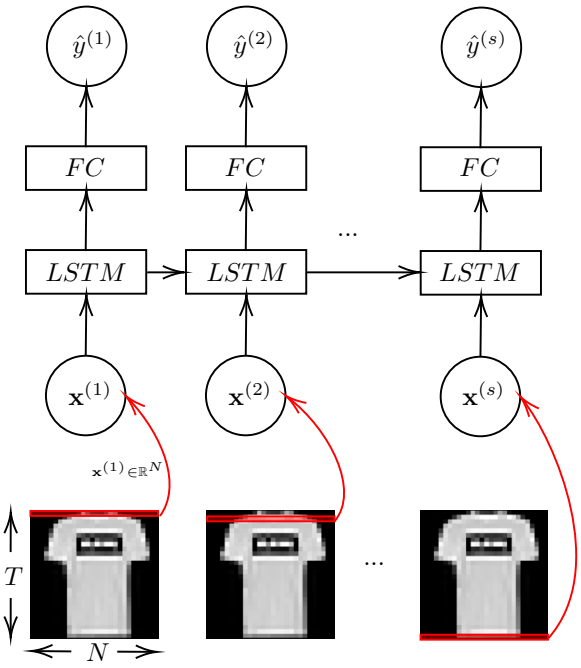


Fig. 2. Arquitectura de la red utilizada.

B. Resultados experimentales

A continuación mostramos los resultados obtenidos tras el cálculo de una media aritmética de diez ejecuciones independientes para cada una de las funciones de agregación así como para las combinaciones lineales de funciones. En las tablas II y III se recalca con negrita el resultado con mayor precisión obtenida.
En primer lugar, en la tabla II mostramos los resultados obtenidos con una sola función de agregación. El mejor resultado promedio lo obtenemos cuando realizamos la agregación de los valores mediante la integral de Choquet, pero cuando el exponente $q > 0$ es aprendido por la propia red neuronal recurrente. Esto supone que el algoritmo modeliza mejor la interacción y posible coalición entre los datos. De esta manera, obtenemos una ponderación de los datos que permite una mejora de 1.01 puntos con respecto a la forma de agregación clásica en esta arquitectura, la suma.

TABLE II
PRECISIÓN DE DISTINTAS AGREGACIONES PARA EL DATASET FASHION-MNIST (MEDIA DE 10 EJECUCIONES)

Agregación	*Precisión*
Max	86.45
Ch_2	85.65
Ch_q	**90.01**
Sum	89.00

Dado que es posible mejorar los resultados mediante la combinación lineal de funciones de agregación, en la tabla III mostramos los resultados obtenidos a partir de la aplicación de estas últimas. A nivel general, en comparación con los resultados obtenidos con la aplicación de una única función de agregación, podemos observar que todos los resultados son mejores, ya que los valores $\lambda_1, \lambda_2 \in \mathbb{R}$ permiten modelar con mayor acierto los datos. Así mismo, con similitud a la tabla II, las agregaciones con las que mejor resultado obtenemos son aquellas las cuales uno de sus componentes es la integral de Choquet en la que se aprende la medida.

TABLE III
PRECISIÓN DE DISTINTAS COMBINACIONES DE AGREGACIONES PARA EL DATASET FASHION-MNIST (MEDIA DE 10 EJECUCIONES)

Combinación	*Precisión*
$Max + Sum$	89.78
$Ch_2 + Sum$	90.04
$Ch_q + Sum$	90.12
$Ch_2 + Max$	89.71
$Ch_q + Max$	**90.14**

VI. CONCLUSIONES

En este trabajo hemos propuesto un nuevo método para la fusión de vectores multidimensionales, así como la utilización de dicha expresión para la fusión de información secuencial en redes neuronales recurrentes tipo LSTM. Así mismo, se ha corroborado una mejora en la precisión en el dataset *Fashion-MNIST*. Hemos observado que los mejores resultados se obtienen cuando sustituimos la suma por la integral de Choquet.
En cuanto a las líneas futuras, en el aspecto teórico nuestra intención es continuar investigando nuevas formas de fusión de vectores basadas en la integral de Choquet, como por ejemplo la generalización de expresiones o la utilización de órdenes admisibles. En la vertiente aplicada, las líneas futuras van en la dirección de la modificación de más arquitecturas y más complejas, así como la utilización de otros conjuntos de datos.

AGRADECIMIENTOS

Este trabajo ha sido financiado por la Agencia Estatal de Investigación (España) bajo el proyecto PID2019-108392GB-I00 (AEI/10.13039/ 501100011033).

REFERENCES

[1] G. Beliakov, A. Pradera, and T. Calvo, "Aggregation functions: A guide for practitioners," in *Studies in Fuzziness and Soft Computing*, 2007.

[2] L. De Miguel, M. Sesma-Sara, M. Elkano, M. Asiain, and H. Bustince, "An algorithm for group decision making using n-dimensional fuzzy sets, admissible orders and owa operators," *Information Fusion*, vol. 37, pp. 126–131, 2017.

[3] M. Grabisch, J.-L. Marichal, R. Mesiar, and E. Pap, *Aggregation Functions*. Encyclopedia of Mathematics and its Applications, Cambridge University Press, 2009.

[4] D. Paternain, J. Fernandez, H. Bustince, R. Mesiar, and G. Beliakov, "Construction of image reduction operators using averaging aggregation functions," *Fuzzy Sets and Systems*, vol. 261, pp. 87–111, 2015. Theme: Aggregation operators.

[5] G. Lucca, J. A. Sanz, G. P. Dimuro, B. Bedregal, M. J. Asiain, M. Elkano, and H. Bustince, "Cc-integrals: Choquet-like copula-based aggregation functions and its application in fuzzy rule-based classification systems," *Knowledge-Based Systems*, vol. 119, pp. 32–43, 2017.

[6] C. A. Dias, J. C. S. Bueno, E. N. Borges, S. Botelho, G. Dimuro, G. Lucca, J. Fernández, H. Bustince, and P. Drews, "Using the choquet integral in the pooling layer in deep learning networks," in *NAFIPS*, 2018.

[7] G. Beliakov, H. Bustince, and T. Calvo, "A practical guide to averaging functions," in *Studies in Fuzziness and Soft Computing*, 2016.

[8] M. Grabisch, J.-L. Marichal, R. Mesiar, and E. Pap, "Aggregation functions: Means," *Inf. Sci.*, vol. 181, pp. 1–22, 2011.

[9] M. Grabisch and C. Labreuche, "A decade of application of the choquet and sugeno integrals in multi-criteria decision aid," *Annals of Operations Research*, vol. 175, pp. 247–286, 2008.

[10] G. Lucca, J. A. Sanz, G. P. Dimuro, E. N. Borges, H. Santos, and H. Bustince, "Analyzing the performance of different fuzzy measures with generalizations of the choquet integral in classification problems," in *2019 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pp. 1–6, 2019.

[11] G. Choquet, "Theory of capacities," *Annales de l'Institut Fourier*, vol. 5, pp. 131–295, 1954.

[12] H. Bustince, R. Mesiar, J. Fernandez, M. Galar, D. Paternain, A. Altalhi, G. Dimuro, B. Bedregal, and Z. Takáč, "d-choquet integrals: Choquet integrals based on dissimilarities," *Fuzzy Sets and Systems*, vol. 414, pp. 1–27, 2021. Aggregation Functions.

[13] G. P. Dimuro, G. Lucca, B. Bedregal, R. Mesiar, J. A. Sanz, C.-T. Lin, and H. Bustince, "Generalized cf1f2-integrals: From choquet-like aggregation to ordered directionally monotone functions," *Fuzzy Sets and Systems*, vol. 378, pp. 44–67, 2020. Theme : Aggregation Operations.

[14] G. Lucca, J. Antonio Sanz, G. P. Dimuro, B. Bedregal, H. Bustince, and R. Mesiar, "Cf-integrals: A new family of pre-aggregation functions with application to fuzzy rule-based classification systems," *Information Sciences*, vol. 435, pp. 94–110, 2018.

[15] A. Graves, *Supervised Sequence Labelling with Recurrent Neural Networks*. Studies in computational intelligence, Berlin: Springer, 2012.

[16] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "Lstm: A search space odyssey," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 10, pp. 2222–2232, 2017.

[17] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[18] T. Murofushi, M. Sugeno, and M. Machida, "Non-monotonic fuzzy measures and the choquet integral," *Fuzzy Sets and Systems*, vol. 64, no. 1, pp. 73–86, 1994.

[19] L. Jin, M. Kalina, R. Mesiar, and S. Borkotokey, "Discrete choquet integrals for riemann integrable inputs with some applications," *IEEE Transactions on Fuzzy Systems*, vol. 26, no. 5, pp. 3164–3169, 2018.

[20] B. Bedregal, G. Beliakov, H. Bustince, T. Calvo, R. Mesiar, and D. Paternain, "A class of fuzzy multisets with a fixed number of memberships," *Information Sciences*, vol. 189, pp. 1–17, 2012.

[21] S. Hochreiter, Y. Bengio, P. Frasconi, and J. Schmidhuber, "Gradient flow in recurrent nets: the difficulty of learning long-term dependencies," in *A Field Guide to Dynamical Recurrent Neural Networks* (S. C. Kremer and J. F. Kolen, eds.), IEEE Press, 2001.

[22] F. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: continual prediction with lstm," in *1999 Ninth International Conference on Artificial Neural Networks ICANN 99. (Conf. Publ. No. 470)*, vol. 2, pp. 850–855 vol.2, 1999.

[23] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms," *ArXiv*, vol. abs/1708.07747, 2017.

[24] Q. V. Le, N. Jaitly, and G. E. Hinton, "A simple way to initialize recurrent networks of rectified linear units," *CoRR*, vol. abs/1504.00941, 2015.

Monotonicity in Non-deterministic Computable Aggregations

1st Luis Magdalena
ETSI Informáticos
Universidad Politécnica de Madrid
Madrid, Spain
0000-0001-7639-8906

2nd Daniel Gómez
Departamento de Estadística y Ciencia de Datos
Complutense University of Madrid
Madrid, Spain
0000-0001-9548-5781

3rd Luis Garmendia
Facultad de Informática
Complutense University of Madrid
Madrid, Spain
0000-0003-3429-8434

4th Javier Montero
Departamento de Estadística e Investigación Operativa
Complutense University of Madrid
Madrid, Spain
0000-0001-8333-2155

***Abstract*—Computable aggregation operators can be seen as a generalization of aggregation operators where the mathematical function is replaced by a program that performs the aggregation process. This extension allows the introduction of new aggregation processes not feasible under the classical framework. Particularly interesting are some non-deterministic processes widely considered to merge information. However, especially in non-deterministic processes, the extension of some of the well-known concepts for aggregation operators such as monotony, is needed. In this work, a new concept of monotonicity is proposed, from a probabilistic perspective, for non-deterministic computable aggregation operators. To be consistent, the concept coincides with the classical definition in the deterministic case. In addition, some cases of interest are analysed.**

***Index Terms*—aggregation, computable aggregation, monotonicity.**

I. Introduction

One of the most important processes for dealing with complex information is the aggregation process [1]–[6].

Aggregation is a key tool for most knowledge-based systems. In general, we can say that aggregation has the aim of using different pieces of information to come to a conclusion or a decision. Several research communities consider these tools, such as the multi-criteria community, the decision-sensor fusion community, the decision-making community, and the data mining community, among many others.

Aggregation functions have been associated in literature with aggregation processes. In this sense, the process to aggregate information has usually been modelled by a function or a family of functions.

Nevertheless, in the pioneering work of Montero [7] the concept of computable aggregations was defined. In that definition, an aggregation process is associated with a program not necessarily being expressed in terms of functions. This approach focuses on the way each aggregation is obtained, i.e., the procedure that produces the aggregated value. Relevant properties come from these procedures, and it is the specific procedure we apply what should be the main object of study.

Furthermore, the rupture between functions and aggregation operators allows to open the domain of aggregation processes to a field not yet analysed in this discipline: non-deterministic computable aggregations [8], which are those aggregations in which it cannot be guaranteed that the results of the information that is going to be aggregated coincide when replicating the process, so they are not functions. This type of aggregation procedure is very common in statistics, where, due to the volume of information that is processed, it is frequent to choose a representative sample on which to operate. Obviously, replicating the process does not imply that the sample coincides and therefore the result varies. This kind of computable aggregation needs to redefine some common properties of aggregation operators, as monotonicity, being no longer valid for non-deterministic computable aggregations.

Monotonicity has been always present in aggregation processes, from the initial ideas by Zadeh (corresponding to set operations as unions and intersections) to more recent proposals (directional monotonicity [6], arity-monotonic aggregation operators [9], ...). Monotonicity seems to be a desirable property associated with a certain robustness of the aggregation process (obviously not always needed). As an example, when aggregating information representing positive evidence on a fact, one expects aggregation being monotone. Even if the aggregation process is non-deterministic (a non-deterministic computable aggregation), one would expect some kind of monotonicity in this process as well.

The purpose of this paper is to revisit and redefine the monotony property for non-deterministic computable aggregations. Section II contains some preliminary knowledge useful to follow the rest of the paper. In Section III we address the key issue of how the output of a non-deterministic computable aggregation can be described. In Section IV we discuss the concept of empirical monotonicity, and population monotonicity, which generalizes monotonicity in the deterministic framework. Comparison between both concepts is developed in Subsection IV-C. Final Section includes an additional analysis

and stresses the interest of our results.

II. Preliminaries

A. Non-deterministic Computable Aggregations

Aggregation operators were initially defined to deal with fuzzy sets [3], [5], [10]–[12], and as a consequence, they were associated with membership functions. So, this is the reason why they were defined as follows:

Definition 1. [2] An aggregation operator is a mapping $Ag : [0,1]^n \to [0,1]$ that satisfies:

1) $Ag(0,0...,0) = 0$ and $Ag(1,1,..,1) = 1$.
2) Ag is monotonic.

It is even possible in some cases to define aggregation processes going beyond functions by considering methods that do not match with the concept of mapping. To analyse this option let us remind the concept of computable aggregation. The main contribution in [7] was to separate the strong association that existed between "aggregation processes" and explicit functions. But to do so let's first rewrite the previous definition in terms of lists as we will use a list notation all trough this paper.

Definition 2. Let $\mathcal{L}$ be the set of non-empty and finite lists of degrees in $[0,1]$. Then an aggregation operator is a mapping $Ag : \mathcal{L} \to [0,1]$ that satisfies:

1) $Ag([0,0...,0]) = 0$ and $Ag([1,1,..,1]) = 1$.
2) Ag is monotonic.

Definition 3. [7] **(Computable aggregation).** Let $L < T >$ be a non-empty and finite list of n elements with type T. A **computable aggregation** P is a program that transforms the list $L < T >$ into an element of T.

Remark 1. *This paper combines computational and mathematical aspects. From a computational point of view, the term* list *is used to refer to the type of data with which our program (Computable aggregation) works. However, from a mathematical approach, even considering that we will maintain the term* list*, these lists should be analysed as* tuples*.*

Given a program, there exist some situations in which it is not possible to build a function associated to it. Those are, for example, the non-deterministic programs that we introduce in the following definition.

Definition 4 (Deterministic program)**.** A program is deterministic, or repeatable, if it produces the very same output when given the same input, no matter how many times it is run.

On the other hand, we have programs that can not be modelled by functions because of the intrinsic definition of function.

Definition 5 (Non-deterministic computable aggregation)**.** A computable aggregation P over the set T is non deterministic if and only if the program implementing it is non deterministic.

Hereafter, we will assume $T = [0,1]$, and consequently, our lists (L) will be defined in $\mathcal{L}$. The obtained results can be easily extended to any lattice totally ordered and with maximum and minimum.

Let us denote by $\mathcal{P}_{\mathcal{D}}$ the class of deterministic computable aggregations and by $\mathcal{P}_{\mathcal{ND}}$ the set of non deterministic computable aggregations.

The non deterministic behaviour of a program can arise from different sources, but we will concentrate now in those aggregation processes involving random or probabilistic decisions. Some elements of $\mathcal{P}_{\mathcal{ND}}$ with this characteristic are the following [8]:

Definition 6. (Probability sampled computable aggregation). Given a value $p \in (0,1]$, and given a family of aggregation operators $\{Ag_n : [0,1]^n \longrightarrow [0,1],\ n \geq 1\}$, let us define the computable aggregation $P_{Ag,p}$ as the three steps program that for a given list $L_1 = [x_1, \ldots, x_n] \in \mathcal{L}$ performs the following actions:

- **Step 1**. Reduce the list L_1 into another list L_2 of lower (or equal) dimension by randomly erasing the elements of the list with probability $1 - p$.
- **Step 2**. If L_2 is empty then return to Step 1.
- **Step 3**. Return the value $Ag_{|L_2|}(L_2)$ if $|L_2| \geq 1$.

Note that the computable aggregation $P_{Ag,p=1}$ is deterministic since $P_{Ag,1}(L_1) = Ag_{|L_1|}(L_1)$ (since in this case $L_2 = L_1$).

Another way to sample a list is by fixing a value k, and randomly selecting k elements from it. Given a list of m elements in $[0,1]$, $L_1 = [x_1, \ldots, x_m] \in \mathcal{L}$, let us denote by Sel_k a program that randomly chooses a sample (without resampling) of k elements of the list if $k < m$, and maintains the same list if $k \geq m$.

Definition 7. (k-sampled computable aggregation). Given a family of aggregation operators $\{Ag_n,\ n \geq 1\}$, the computable aggregation $P_{Ag,k}$, is defined as the program that for a given list L_1 of m elements, first applies the procedure $L_2 = Sel_k(L_1)$, and then computes the value $Ag_{|L_2|}(L_2)$.

It is important to notice that in the previous definitions the sampled list L_2 should be a sublist of L_1, then maintaining the relative positions among elements.

B. Distribution functions

In our analysis based on populations we will use distribution functions and will consider some related concepts.

Definition 8. Given a random distribution X, the **cumulative distribution function** of X (denoted by F_X), is a function that for each value x_0 computes the probability induced by X of the set $\{X \leq x_0\}$. Formally the function $F_X : \mathbb{R} \longrightarrow [0,1]$, is defined as

$$F_X(x_0) = P(X \leq x_0).$$

We can simply use the term *distribution function* to refer to the *cumulative distribution function*.

Another concept that we need to introduce is the idea of empirical distribution. The empirical distribution function can be understood as an estimation of the cumulative distribution

function that is obtained from a sample of the global population.

Definition 9. Given a list $l = [a_1, \ldots, a_m]$ with m values in $[0, 1]$, then the empirical distribution function associated to l and denoted by EF_l can be formally defined for all $x \in \mathbb{R}$ as

$$EF_l(x) = \frac{|\{i \mid a_i \leq x\}|}{m}.$$

Given a distribution X, if we obtain $l = [a_1, \ldots, a_m]$ in an independent way from the global distribution X, there exist many statistical results that deal with the convergence of the empirical distribution function to the underlying cumulative distribution function. So, in general, when m increases, EF_l converges to F_X.

In addition, given two distributions we can compare them.

Definition 10. Given two distributions X_1 and X_2 with distribution functions F_{X_1} and F_{X_2}: $\mathbb{R} \longrightarrow [0, 1]$. We will say that $X_1 \leq_D X_2$ when for all $x \in \mathbb{R}$, $F_{X_1}(x) \geq F_{X_2}(x)$.

III. Characterising the output of a non deterministic computable aggregation

Taking into account that the core properties of aggregation operators are monotonicity and boundary conditions, it is obvious that defining a new kind of aggregation, as the non deterministic computable aggregations, requires the analysis of this two properties.

The boundary conditions are quite simple concepts that have a straightforward counterpart in this kind of non deterministic computable aggregations. Any sampling of a list of 0s will be a new list of 0s (similarly for a list of 1s), so, as the underlying aggregation satisfies the boundary conditions, the computable aggregation also does. But monotonicity is a quite different question. The analysis of monotonicity requires the comparison of the outputs of two aggregations. The problem is that, differently from other aggregation processes, the output of a non-deterministic aggregation cannot be described in terms of a known and fixed value. This situation rises a previous question prior to consider monotonicity. How can we describe the output of a non-deterministic computable aggregation?

Given a computable aggregation P aggregating a list $L_1 \in \mathcal{L}$, we could describe its output by running the program several times and compiling the outputs. Clearly, if considering a deterministic computable aggregation, all outputs should be the same. However, a non-deterministic computable aggregation would probably produce several different values.

From now on we will denote by $LP^m_{L_1} \in \mathcal{L}$ the list of length m obtained after m executions of the program P over the list L_1. These m-realizations could be characterized in terms of a distribution [13].

Definition 11. (Empirical distribution of a computable aggregation). Given a computable aggregation P and given a list $L_1 \in \mathcal{L}$, the distribution of results obtained after m executions of the program P over the list L_1, will be referred as the *empirical distribution with size m of the program P over the list L_1 in $\mathcal{L}$*, represented by $DP^m_{L_1}$.

It is also possible that, by simply analysing the program or the underlying algorithm, we could determine what should theoretically be the distribution of outputs for a given input.

Given a computable aggregation P that aggregates a list $L_1 \in \mathcal{L}$, let us denote by $\mathcal{P}(L_1)$ the theoretical distribution after all possible realizations of the program P over the fixed list L_1. Obviously, if $P \in \mathcal{P}_{\mathcal{D}}$, the associate $\mathcal{P}(L_1)$ for any list will be a single value. For non deterministic programs, we will have here a probability distribution $\mathcal{P}(L_1)$ for each fixed value of L_1. However there will be some cases where we would be not able to obtain that theoretical distribution.

In general, given a non deterministic computable aggregation P, it is not possible to know the theoretical distribution $\mathcal{P}(L_1)$. Nevertheless, we could try to approximate it by making many realizations of $P(L_1)$.

After defining $LP^m_{L_1}$, $DP^m_{L_1}$ and $\mathcal{P}(L_1)$ we need to consider its relations. In order to show the differences between the list of executions, the empirical distribution, and the theoretical distribution, let's consider the following example.

Example 1. Let $L_1 = [0.1, 0.2, 0.6, 0.7]$ be a list of four elements in $[0, 1]$, and let P be a deterministic program that calculates the average of the elements of the list. Let $m = 5$ be the number of executions. In one hand, it is easy to see that $LP^5_{L_1} = [0.4, 0.4, 0.4, 0.4, 0.4]$ (i.e. a list of 5 elements each one of them taking the value 0.4), since the program is deterministic and the program will give always the same result. On the other hand, the empirical and theoretical distributions $DP^5_{L_1}$ and $\mathcal{P}(L_1)$ coincide with the degenerate distribution in the point 0.4 (i.e. $DP^5_{L_1} = \mathcal{P}(L_1) = \{0.4\}$) that can be described by the following cumulative distribution function.

$$F(x) = \begin{cases} 0 & \text{If } \ x < 0.4 \\ 1 & \text{If } \ x \geq 0.4. \end{cases}$$

We have obtained the list of executions, the empirical distribution and the theoretical distribution.

Once analysed the previous example considering a deterministic computable aggregation, let's consider now a generic approach including the non deterministic case. Taking into account the concept of distribution function presented in previous section, if we want to estimate the distribution of $\mathcal{P}(L_1)$, we can execute m times $P(L_1)$, obtaining the list $LP^m_{L_1}$. From this list, it is possible to obtain what we have denoted by $DP^m_{L_1}$ that is the empirical distribution of the values of the list $LP^m_{L_1}$. The result will be:

$$F_{DP^m_{L_1}} = EF_{LP^m_{L_1}}. \tag{1}$$

Proposition 1. *[14] Given a deterministic computable aggregation P, the following holds:*

$$DP^m_{L_1} = \mathcal{P}(L_1).$$

Starting from a list $L_1 \in \mathcal{L}$ of elements to be aggregated, and a non deterministic aggregation process, we have defined two distributions describing the result of the aggregation process: the theoretical distribution $\mathcal{P}(L_1)$ and the empirical distribution ($DP^m_{L_1}$). In addition we have the list $LP^m_{L_1}$

obtained after m executions of the aggregation process. The interactions and relations among these elements will describe and characterize a non deterministic computable aggregation.

IV. Monotonicity in non-deterministic computable aggregation

In a first approach to the monotonicity condition, we will assume that the output for a certain input list $L_1 \in \mathcal{L}$ is described in terms of m-realizations $LP^m_{L_1} \in \mathcal{L}$, being also a list. In this situation any monotonicity analysis will require a method to compare/order elements of $\mathcal{L}$.

Let $L_1 = [a_1, \ldots, a_n]$ and $L_2 = [b_1, \ldots, b_n]$ be two lists of $\mathcal{L}$ with the same length $n > 0$. We will represent by $s(L_1)$ the list obtained by sorting L_1 in increasing order. Accordingly, $s(L_2)$ will correspond to the sorted version of L_2.

Definition 12. (Sorted 1to1 preorder). A list L_1 in $\mathcal{L}$ is S1to1 lower or equal than a list L_2 in $\mathcal{L}$, if and only if the kth value of $s(L_1)$ is lower or equal to the kth value of $s(L_2)$, for all k from 1 to n, that is:

$$L_1 \leq_{\text{S1to1}} L_2 \text{ if and only if } s(L_1)_k \leq s(L_2)_k, \forall k = 1, \ldots n.$$

Definition 13. (1to1 partial order). A list $L_1 = [a_1, \ldots, a_n]$ in $\mathcal{L}$ is 1to1 lower or equal than a list $L_2 = [b_1, \ldots, b_n]$ in $\mathcal{L}$, if and only if the kth element of L_1 is lower or equal than the kth degree of L_2. It is:

$$L_1 \leq_{\text{1to1}} L_2 \text{ if and only if } a_k \leq b_k, \forall k = 1, \ldots n.$$

This partial order is equivalent to the one obtained by considering the lists (of lenght n) in $\mathcal{L}$ as elements in $\mathbb{R}^n$ with its usual order.

A. Empirical monotonicity for non-deterministic computable aggregations

How to generalize the idea of monotonicity from classical aggregation operators to non-deterministic computable aggregations is not a trivial task. In general terms, monotonicity implies that if we have two input lists $L_1 \leq L_2$, their outputs should maintain that order. Consequently, the usual concept of monotonicity of aggregation operators is not valid for non-deterministic computable aggregations, as these aggregations do not produce the same output when receiving the same input. To cope with this situation, the concept of list associated to m executions for a given list L_i ($LP^m_{L_i}$) has been previously introduced. This concept describes the result obtained (a list $L_o \in \mathcal{L}$) after m executions of the program P over the list L_i. When m is large enough it allows an empirical analysis of the computable aggregation.

The following properties try to answer the question of how to define monotonicity from this point of view, applying the different methods previously considered for the comparison of lists. It is important to notice that we have lists as inputs (L_i), and we describe the outputs also in terms of lists ($LP^m_{L_i}$).

In what concerns the lists of inputs (L_i), from the point of view of an aggregation processes these lists behave as vectors (the n inputs of an aggregation process are an element of $[0,1]^n$), and should be ordered considering the usual order of $[0,1]^n$, that is equivalent to $\leq_{\text{1to1}}$ previously defined. Consequently we will use this notation in the definitions.

Definition 14. (Strong $\leq$-monotonicity of non-deterministic computable aggregations]). Let P be a non-deterministic computable aggregation. Let $L_1 = [a_1, \ldots, a_n]$ and $L_2 = [b_1, \ldots, b_n]$ be two lists of degrees in $\mathcal{L}$. Let $\leq$ be a (partial) order on the set $\mathcal{L}$. A non-deterministic computable aggregation P is strong $\leq$-monotone if and only if, when $L_1 \leq_{\text{1to1}} L_2$ then $(LP^m_{L_1}) \leq (LP^m_{L_2})$ for any $m \geq 1$.

Definition 15. (Asymptotic $\leq$-monotonicity of non-deterministic computable aggregations). Let P be a non-deterministic computable aggregation. Let $L_1 = [a_1, \ldots, a_n]$ and $L_2 = [b_1, \ldots, b_n]$ be two lists of degrees in $\mathcal{L}$. Let $\leq$ be a partial order on the set $\mathcal{L}$. A non-deterministic computable aggregation P is asymptotic $\leq$-monotone if and only if, there exists a natural number n_0 such that If $L_1 \leq_{\text{1to1}} L_2$ then $(LP^m_{L_1}) \leq (LP^m_{L_2})$ for any $m > n_0$.

B. Population monotonicity for non-deterministic computable aggregations

Once defined an order in the output-space of non deterministic computable aggregations, let us consider monotonicity.

Definition 16 (Population monotonicity of non-deterministic computable aggregations)**.** Let P be a non-deterministic computable aggregation, and let L_1 and L_2 be two lists $\in \mathcal{L}$ ordered with the classical vectorial order definition (equivalent to the previously defined $1to1$ order). A non-deterministic computable aggregation P is **population monotone** if and only if, when $L_1 \leq L_2$ then $\mathcal{P}(L_1) \leq_D \mathcal{P}(L_2)$.

Remark 2. *Let us observe that previous definition generalizes the classical definition of monotonicity for deterministic computable aggregations. In the following proposition we will see in detail.*

Proposition 2. *Let P be a deterministic computable aggregation with associated function Ag. Then, the following holds:*

P is population monotone if and only if Ag is monotone.

Proof. • From left to right. Given $L_1 \leq L_2$, will prove that $Ag(L_1) \leq Ag(L_2)$. First, since P is deterministic with associated aggregation function Ag, the following holds for the list L_1 and L_2.

$$F_{\mathcal{P}(L_1)}(x) = \begin{cases} 0 & \text{If } x < Ag(L_1) \\ 1 & \text{If } x \geq Ag(L_1) \end{cases}$$

and

$$F_{\mathcal{P}(L_2)}(x) = \begin{cases} 0 & \text{If } x < Ag(L_2) \\ 1 & \text{If } x \geq Ag(L_2). \end{cases}$$

Then, since P is monotone, $\mathcal{P}(L_1) \leq_D \mathcal{P}(L_2)$ which implies that for all x $F_{\mathcal{P}(L_1)}(x) \geq F_{\mathcal{P}(L_2)}(x)$. Finally, it is very easy to see that these inequalities hold if and only if $Ag(L_1) \leq Ag(L_2)$ so the result is proved.

- From right to left. Given $L_1 \leq L_2$, we are going to prove that $\mathcal{P}(L_1) \leq_D \mathcal{P}(L_2)$. Since P is deterministic, $\mathcal{P}(L_1)$ and $\mathcal{P}(L_2)$ are degenerated distributions in the values $Ag(L_1)$ and $Ag(L_2)$, respectively. Since Ag is monotone, $Ag(L_1) \leq Ag(L_2)$, and thus it is very easy to see that $F_{\mathcal{P}(L_2)}(x) \leq F_{\mathcal{P}(L_1)}(x)$ for any value of x.

□

For practical reasons, we can approximate the theoretical distribution empirically if the number of executions is enough to ensure that empirical distribution is close to theoretical distribution. Taking into account this consideration, we present the following definition.

Definition 17 (Population asymptotic monotonicity of non-deterministic computable aggregations)**.** Let P be a non-deterministic computable aggregation, and let L_1 and L_2 be two lists in $\mathcal{L}$ ordered with the classical vectorial order definition. A non-deterministic computable aggregation P is **population asymptotic monotone** if and only if when $L_1 \leq L_2$ then, $\forall x \in R$,

$$\lim_{m \longrightarrow \infty} \left(F_{DP^m_{L_1}}(x) - F_{DP^m_{L_2}}(x) \right) \geq 0.$$

Proposition 3. *Let P be a non deterministic computable aggregation, if P is* **population monotone** *then P is* **population asymptotic monotone**.

Proof. First of all, let us note that $F_{DP^m_{L_1}}$ and $F_{DP^m_{L_2}}$ are bounded functions in $[0, 1]$. Also, by Glivenko-Cantelli theorem, there exist punctual convergence of the empirical distribution function ($F_{DP^m_L}$) into the theoretical one (denoted as $F_{P(L)}$) for $L \in \{L_1, L_2\}$. Consequently, the limits of $F_{DP^m_{L_1}}(x)$ and $F_{DP^m_{L_2}}(x)$ exist for all $x \in R$, and the limit of the difference coincides with the difference of limits. So, the following holds:

$$\lim_{m \longrightarrow \infty} \left(F_{DP^m_{L_1}}(x) - F_{DP^m_{L_2}}(x) \right) = \lim_{m \longrightarrow \infty} F_{DP^m_{L_1}}(x) - \lim_{m \longrightarrow \infty} F_{DP^m_{L_2}}(x)$$

Now, due to the punctual convergence previously mentioned,

$$\lim_{m \longrightarrow \infty} F_{DP^m_{L_1}}(x) - \lim_{m \longrightarrow \infty} F_{DP^m_{L_2}}(x) = F_{P_{L_1}}(x) - F_{P_{L_2}}(x).$$

Finally, since P is population monotone, $P(L_1) \leq_D P(L_2)$ so the previous difference is greater than or equal to zero and the result is proved.

□

C. Empirical and population monotonicity

Finally, in this subsection we will see the relationship between the defined concepts of monotonicity for lists and distributions. In particular, let us analyse the relations between population monotonicity, asymptotic population monotonicity, asymptotic $\leq$ monotonicity and strong $\leq$ monotonicity.

Let us first consider the relation between the order $\leq_D$ for distribution functions and the order $\leq_{S1to1}$ for lists.

Proposition 4. *Given a computable aggregation P, two lists $L_1, L_2 \in \mathcal{L}$, two m-realizations of the computable aggregation on these lists $LP^m_{L_1}, LP^m_{L_2} \in \mathcal{L}$, and the corresponding empirical distributions $DP^m_{L_1}$ and $DP^m_{L_2}$, then $DP^m_{L_1} \leq_D DP^m_{L_2}$ if and only if $LP^m_{L_1} \leq_{S1to1} LP^m_{L_2}$.*

It is relevant to mention that the strong monotonicity defined for lists will not have an equivalence in the context of distributions since the requirements are too strong, so we will focus on the equivalence in the asymptotic cases for lists and distributions. Our first result is a natural consequence of Proposition 4, which establishes an equivalence between the orders in distributions after m realizations and the associated lists.

Proposition 5. *P is population asymptotic monotone if P is $\leq_{S1to1}$-asymptotic monotone.*

Proof. This is direct by Proposition 4 in which we have that $DP^m_{L_1} \leq_D DP^m_{L_2}$ if and only if $LP^m_{L_1} \leq_{S1to1} LP^m_{L_2}$. So if P is $\leq_{S1to1}$-asymptotic monotone, then $DP^m_{L_1} \leq_D DP^m_{L_2}$ for a given value of $m \geq m_0$, and as a consequence $\lim_{m \longrightarrow \infty} \left(F_{DP^m_{L_1}}(x) - F_{DP^m_{L_2}}(x) \right) \geq 0$. □

Now taking into account the previous implications between list of orders, distributions, and different classes of monotonicity, the following implications trivially holds.

Corollary 1. *Let P be a computable aggregation.*

- *If P is strong $\leq_{S1to1}$ monotone, then P is population asymptotic monotone.*
- *If P is strong $\leq_{S1to1}$ monotone, then P is population monotone.*
- *If P is a deterministic computable aggregation, then P is strong $\leq$ monotone, for any order.*

Prior to conclude the paper with a preliminary analysis on sampling aggregations, it is important to remark the reason to introduce the analysis based on populations, once we have the empirical approach. The main reason relies in the fact that once we move from lists to distributions, the dimension of the list does not affect. We can compare distributions related to lists of different dimension. That is, we can compare $DP^m_{L_1}$ and $DP^n_{L_2}$, no matter if $n \neq m$.

V. Final analysis and conclusions

Once introduced the different definitions and properties, it is possible to analyse monotonicity for a family of non deterministic computable aggregations that includes the most famous non deterministic process in statistics: sampling. We will consider now the k-sampled and probability sampled computable aggregations.

It is possible to check that both families of non deterministic computable aggregations are population monotone, for any classical aggregation operator function Ag. To do so, a possible guideline of the proof is to use the concept of empirical

distribution function associated to a list (EF_l, see definition 9) and the distribution associated to a list l denoted as D_l, and described (according to Eq. 1) as $F_{D_l} = EF_l$. Note that this is a generalization of the relation defined between a list of $m-$realizations ($LP^m_{L_1}$) and the corresponding empirical distribution ($DP^m_{L_1}$).

The idea beyond this, is to compare the associated empirical distribution with the population distributions associated with lists that are generated after we aggregate all possible scenarios of these two computable aggregations.

To conclude this section, let us note that in this work we provide an approach to monotonicity of non deterministic computable aggregations. This definition includes the classical notion in the deterministic case and offers two different approaches. The first approach relies on orders (and preorders) on lists. The second represents a probabilistic conception appearing in a natural way since we model the outputs of a computable aggregation (that in the deterministic case are single points), as probability distributions. As a consequence, we have had to answer the question of how to order lists and/or probability distributions. In this work, we provide a natural way of comparing distributions following the idea of stochastic dominance between distribution functions.

To conclude we make two additional comments. On the one hand, it is clear that other definitions for ordering lists and probability distributions could be considered and would give different versions of monotonicity. How to establish other orders between lists and between probability distributions is a question that we believe is worth studying in the future. On the other hand, we have also interpreted the output of a computable aggregation after a sufficient number of iterations, simply as a list of values. In that framework, we can also explore other order relations (now between lists) that would produce different notions of monotonicity.

ACKNOWLEDGMENT

This research has been partially supported by the Government of Spain (grant PGC2018-096509-B-I00), Comunidad de Madrid (grant S2013/ICCE-2845 and Convenio Plurianual con la Universidad Politécnica de Madrid en la línea de actuación Programa de Excelencia para el Profesorado Universitario), Complutense University (UCM Research Group 910149), and Universidad Politécnica de Madrid.

REFERENCES

[1] B. Bouchon-Meunier, *Aggregation and fusion of imperfect information*. Physica, 2013, vol. 12.

[2] G. Beliakov, A. Pradera, and T. Calvo, *Aggregations Functions: A guide for Practitioners*. Springer, 2007.

[3] G. Beliakov, D. Gómez, S. James, . Montero, and J. Rodríguez, "Approaches to learning strictly-stable weights for data missing values," *Fuzzy Sets and Systems*, vol. 325, pp. 97–113, 2017.

[4] D. Gómez and J. Montero, "A discussion on aggregations operators," *Kybernetika*, vol. 40, pp. 107–120, 2004.

[5] D. Gómez, K. Rojas, J. Montero, J. Rodríguez, and G. Beliakov, "Consistency and stability in aggregation operators, an application to missing data problems," *International Journal of Computational Intelligence Systems*, vol. 7, pp. 595–604, 2014.

[6] M. Sesma-Sara, R. Mesiar, and H. Bustince, "Weak and directional monotonicity of functions on riesz spaces to fuse uncertain data," *Fuzzy Sets and Systems*, vol. 386, pp. 145–160, 2020, aggregation Operations.

[7] J. Montero, R. G. del Campo, L. Garmendia, D. Gómez, and J. Rodríguez, "Computable aggregations," *Information Sciences*, vol. 460, pp. 439–449, 2018.

[8] L. Garmendia, D. Gómez, L. Magdalena, and J. Montero, "Analyzing non-deterministic computable aggregations," in *Information Processing and Management of Uncertainty in Knowledge-Based Systems*, M.-J. Lesot, S. Vieira, M. Z. Reformat, J. P. Carvalho, A. Wilbik, B. Bouchon-Meunier, and R. R. Yager, Eds. Cham: Springer International Publishing, 2020, pp. 551–564.

[9] M. Gagolewski and P. Grzegorzewski, "Arity-monotonic extended aggregation operators," in *Information Processing and Management of Uncertainty in Knowledge-Based Systems. Theory and Methods*, E. Hüllermeier, R. Kruse, and F. Hoffmann, Eds. Springer Berlin Heidelberg, 2010, pp. 693–702.

[10] T. Calvo, A. Kolesárová, M. Komorníková, and R. Mesiar, "Aggregation operators: properties, classes and constructions methods," in *Aggregation Operators*, ser. Studies in Fuzziness and Soft Computing, C. T., M. G., and M. R., Eds. Springer, 2002, vol. 97, pp. 3–104.

[11] K. Rojas, D. Gómez, J. Montero, and J. Rodríguez, "Strictly stable families of aggregation operators," *Fuzzy Sets and Systems*, vol. 228, pp. 44–63, 2013.

[12] P. Olaso, K. Rojas, D. Gómez, and J. Montero, "A generalization of stability for families of aggregation operators," *Fuzzy Sets and Systems*, 2019.

[13] L. Garmendia, D. Gómez, L. Magdalena, and J. Montero, "Empirical monotonicity of non-deterministic computable aggregations," in *The 19th World Congress of the International Fuzzy Systems Association (IFSA-2021)*, 2021.

[14] L. Magdalena, D. Gómez, L. Garmendia, and J. Montero, "Population monotonicity of non-deterministic computable aggregations," in *2021 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, 2021.

Optimizando Desviaciones Moderadas Ponderadas para Interfaces Cerebro Ordenador

1st Javier Fumanal-Idocin
Estadística, Informática y Matemáticas
Universidad Pública de Navarra
Pamplona, Spain
javier.fumanal@unavarra.es

2nd Carmen Vidaurre
Estadística, Informática y Matemáticas
Universidad Pública de Navarra
Pamplona, Spain
carmen.vidaurre@unavarra.es

3rd Marisol Gomez
Estadística, Informática y Matemáticas
Universidad Pública de Navarra
Pamplona, Spain
marisol@unavarra.es

4th Asier Urio
Estadística, Informática y Matemáticas
Universidad Pública de Navarra
Pamplona, Spain
asier.urio@unavarra.es

5th Graçaliz Pereira Dimuro
Informática e Matemática Aplicada
Universidade do Rio Grande do Norte
Natal, Brazil
gracalizdimuro@furg.br

6th Humberto Bustince
Estadística, Informática y Matemáticas
Universidad Pública de Navarra
Pamplona, Spain
bustince@unavarra.es

***Abstract*—Las interfaces cerebro-ordenador (BCI) basadas en el análisis de Electroencefalografía (EEG) están compuestas por varios elementos para procesar y clasificar las señales de entrada del cerebro. Una fase relevante de estos sistemas es el módulo de toma de decisiones, en el que la salida de diferentes clasificadores se fusiona en uno solo. En este trabajo proponemos el uso de funciones basadas en desviaciones moderadas con ponderaciones para la fase de toma de decisiones del sistema de BCI de fusión multimodal mejorado (EMF). Las funciones de agregación basadas en desviación moderada (MD) nos permiten elegir el mejor valor para agregar un vector de puntos utilizando una función de desviación moderada. Usando una MD ponderada, también podemos tener en cuenta la importancia relativa de cada dimensión en los datos multidimensionales que estamos agregando. Utilizando estas funciones en el EMF, podemos ponderar cada una de las diferentes señales cerebrales según su importancia, y utilizando la diferenciación automática, también podemos optimizarlas para el problema concreto a solucionar.**

I. Introducción

Las interfaces cerebro-ordenador (BCI) tienen como objetivo decodificar patrones de señales cerebrales para controlar diferentes mecanismos del cerebro [1], [2]. Se pueden utilizar diferentes propiedades de una señal para controlar dispositivos mediante señales cerebrales, [3], [4]. Una estrategia popular para modular las señales cerebrales de modo que se puedan interpretar en comandos es el entrenamiento con imágenes motoras (MI). Durante el proceso de MI, una persona imagina el movimiento de una parte del cuerpo, como por ejemplo la mano izquierda o derecha, los pies o la lengua [5]. Durante la imaginación de los movimientos, la potencia eléctrica de las áreas sensoriomotoras, contralaterales al lado del movimiento, se reduce, en un efecto conocido como desincronización relacionada con el evento (ERD), [6]–[8] y, a menudo, también se da un aumento de la potencia, o sincronización relacionada con el evento (ERS) en el lado ipsilateral. La identificación correcta de ERD/ERS influye en gran medida en el rendimiento de un sistema BCI basado en MI. Algunos algoritmos muy populares para identificar y clasificar los cambios de potencia de las señales MI son el patrón espacial común (CSP), las máquinas vectoriales de soporte o las redes neuronales de aprendizaje profundo, [6], [9]–[14].

Los sistemas BCI se diseñan utilizando una amplia gama de técnicas para extraer características de los datos originales en bruto antes de extraer patrones cerebrales clasificables. Algunos procedimientos comunes incluyen la estimación de la potencia de banda estrecha en el dominio de tiempo [15], [16], en el dominio de frecuencia [17]–[19] o en el dominio de tiempo-frecuencia utilizando, por ejemplo, ondas de Meyer [20], [21]. Posteriormente, la clasificación se realiza generalmente empleando clasificadores lineales como el Análisis Discriminante Lineal (LDA), pero también QDA o SVMs [22] son procedimientos de clasificación populares. A veces, especialmente en el caso de paradigmas multiclase, están involucradas muchas características diferentes o se combinan diferentes características. En esos casos, el módulo de reconocimiento de patrones puede estar compuesto por un conjunto de clasificadores. La estrategia más común para combinar resultados de clasificación es probablemente la votación por mayoría [23].

En [24], los autores propusieron una nueva forma de realizar la toma de decisiones en un marco BCI en dos fases, la toma de decisiones multimodal. En [25] los autores propusieron una actualización de este mismo proceso, lo que resultó en el marco BCI de fusión multimodal mejorado (EMF). Este trabajo también presentó un estudio detallado de diferentes funciones de agregación, que mostró diferencias significativas en cuanto a la precisión de la clasificación final.

Sin embargo, el problema de elegir la mejor función de agregación sigue estando abierto. La literatura sobre teoría de la agregación sugiere el uso de agregaciones basadas en desviaciones moderadas (MD) como una opción para resolver el problema de escoger la agregación más adecuada para conjunto de datos [26]. La idea de las MD es utilizar una desviación moderada para calcular la similitud entre el posible valor de salida y el conjunto de datos de entrada. Luego,

buscamos el valor que maximiza esa similitud con respecto a los datos de entrada.

En este trabajo, usamos la MD desarrollada en [27] para tomar en cuenta las diferentes importancias de cada característica de entrada usando pesos, y para resolver el problema de elegir la mejor agregación. Mostramos cómo se aplican estas funciones al marco BCI publicado en [25] y un procedimiento para aprender los pesos de acuerdo con los datos de entrada.

El resto del artículo es el siguiente: en la sección II explicamos las funciones basadas en la desviación moderada multivariante ponderada (II-A) y el marco de fusión multimodal mejorado (II-B) y como aplicar estas funciones en dicho marco. En la Sección III describimos nuestros experimentos y mostramos los resultados obtenidos, y en la Sección IV damos nuestras observaciones finales para este trabajo y explicamos nuestras pautas futuras.

II. Métodos

En esta sección explicamos los conceptos asociados con las funciones basadas en la MD y el EMF para la clasificación de señales MI-BCI.

A. Desviaciones moderadas

Sea I un intervalo cerrado de números reales y sea $a = \min I$ y $b = \max I$. Una aplicación $M\colon\ I^n \to I$ se dice que es una función de agregación *(n-dimensional)* si M es creciente y $M(a,\dots,a) = a$, $M(b,\dots,b) = b$. Además, se dice que una función de agregación M es *promedia* [28] si solo si

$$\min(x_1,\dots,x_n) \le M(x_1,\dots,x_n) \le \max(x_1\dots,x_n)$$

por cada $x_1,\dots,x_n \in I$.

Se puede encontrar una lista completa de varios ejemplos de funciones de agregación en [29]. Existen muchas familias de funciones de agregación, utilizadas según las circunstancias de cada problema. Entre ellas, el concepto de agregación basada en desviación resulta muy eficaz en muchas circunstancias. Su procedimiento es sencillo: se mide la diferencia entre los valores a agregar y un posible valor de salida y se escoge el valor que sea más similar a todos ellos.

Para medir la disimilitud en estos pares de valores, se puede utilizar una función de desviación o una función de desviación moderada. Sea I un intervalo cerrado de números reales. Se dice que un mapeo $D\colon\ I^2 \to \mathbb{R}$ es una *función de desviación* si y solo si $D(x,\cdot)\colon\ I \to \mathbb{R}$ es continuo y estrictamente creciente, y $D(x,x) = 0$ para todo $x \in I$. Además, sea n un número natural, un mapeo $M_D\colon\ I^n \to \mathbb{R}$ se dice que es una *media de Daróczy* si y solo si M_D viene dado por

$$M_D(\mathbf{x}) = y,$$

donde y es la solución de la ecuación

$$\sum_{i=1}^{n} D(x_i, y) = 0.$$

Una media de Daróczy nos permite determinar qué tan diferente es cada entrada x_i de y, y el resultado final de la función M_D es ese y para el cual la suma de todas las diferencias es tan pequeña como sea posible. Sin embargo, debido a algunos problemas de esta definición, se propusieron las desviaciones moderadas:

Sea I un intervalo cerrado de números reales. Se dice que un mapeo $D\colon\ I^2 \to \mathbb{R}$ es una *función de desviación moderada* si y solo si

1) Por cada $x \in I$, $D(x,\cdot)\colon\ I \to \mathbb{R}$ no es decreciente;
2) Por cada $y \in I$, $D(\cdot,y)\colon\ I \to \mathbb{R}$ no es creciente;
3) $D(x,y) = 0$ si y solo si $x = y$, $x,\ y \in I$.

Una función dada por $D(x,y) = y - x$ para $x, y \in \mathbb{R}$, representa el ejemplo prototípico de función de desviación moderada.

Ahora procedemos a presentar un método para construir desviaciones moderadas ponderadas para datos multi-variable, que luego usaremos para agregar preferencias de distintos expertos.

Proposition II.1. *Sea $f\colon\ \mathbb{R} \to \mathbb{R}$ una función no decreciente tal que $f(x) = 0$ si y solo si $x = 0$. Sea $s\colon\ [0,1] \to \mathbb{R}$ una función estrictamente creciente. Luego, un mapeo $D_{f,s}\colon [0,1]^2 \to \mathbb{R}$ dado por*

$$D_{f,s}(x,y) = f\big(s(y) - s(x)\big)$$

es una función de desviación moderada.

Sea el conjunto de datos multidimensionales una matriz $p \times q$ $\mathbb{A}$ de n vectores de manera que tanto p como q sean divisibles por un entero positivo r, $r \neq 1$. La matriz $\mathbb{A}$ puede considerarse compuesta de n submatrices $\mathbb{A}_1$, $\mathbb{A}_2$, $\dots$, $\mathbb{A}_n$ con entradas reales, donde $\mathbb{A}_j$ es una " sección transversal " de $\mathbb{A}$ a través de todos sus j -ésimos componentes . Por tanto, la matriz $\mathbb{A}$ puede considerarse como el vector $(\mathbb{A}_1, \mathbb{A}_2, \dots, \mathbb{A}_n)$ de matrices.

Descompongamos la matriz $\mathbb{A}$ en $(p/r)\cdot(q/r)$ mutuamente disjuntas submatrices $r \times r$ $\mathbb{B}^{\alpha\beta}$, $\alpha \in \{1,2,\dots,p/r\}$, $\beta \in \{1,2,\dots,q/r\}$. Al aplicar el método de fusión en cuestión, reemplazaremos cada submatriz de n-tuplas por un representante apropiado que es nuevamente una n-tupla, lo que da como resultado un $(p/r) \times (q/r)$ matriz $\mathbb{C}$ de n reales -tuplas.

Paso #1 (Descomposición de la matriz A): sea $\mathcal{I} = [p/r]$ y $\mathcal{J} = [q/r]$. Denotando el elemento en la posición (r,s) de $\mathbb{A}$ por a_{rs}. Se define para cada $\alpha \in \mathcal{I}$ y $\beta \in \mathcal{J}$ una $r \times r$ matriz $\mathbb{B}^{\alpha\beta}$ de n-tuplas como

$$b_{i,j}^{\alpha\beta} = a_{(\alpha-1)r+i,(\beta-1)r+j} \quad \text{para} \quad i,\ j \in [r]. \tag{1}$$

Entonces, la matriz $\mathbb{A}$ puede verse como una unión disjunta de matrices $\mathbb{B}^{\alpha\beta}$ sobre todos los índices $\alpha \in \mathcal{I}$, $\beta \in \mathcal{J}$.

Sea α un índice arbitrario pero fijo de $\mathcal{I}$ y β de $\mathcal{J}$, respectivamente. Entonces la matriz $\mathbb{B}^{\alpha\beta}$ consta de n de $r \times r$

matrices cuadradas

$$\mathbb{B}_k^{\alpha\beta} = \begin{pmatrix} b_{11k}^{\alpha\beta} & b_{12k}^{\alpha\beta} & \cdots & b_{1rk}^{\alpha\beta} \\ b_{21k}^{\alpha\beta} & b_{22k}^{\alpha\beta} & \cdots & b_{2rk}^{\alpha\beta} \\ \vdots & \vdots & \ddots & \vdots \\ b_{r1k}^{\alpha\beta} & b_{r2k}^{\alpha\beta} & \cdots & b_{rrk}^{\alpha\beta} \end{pmatrix} \text{ para cada } k \in [n]. \quad (2)$$

Nota: Dentro del espacio euclidiano n-dimensional $\mathbf{E}^n$ alcance, los elementos de $\mathbb{B}_k^{\alpha\beta}$ corresponden a las proyecciones sobre el k-eje.

Paso #2 (Fijando un dominio): teniendo en cuenta las posibles aplicaciones, es necesario fijar un intervalo real cerrado para ser utilizado como dominio de una función de agregación basada en desviaciones. Por esta razón, designamos

$$\wedge_k^{\alpha\beta} = \min_{i,j\in[r]}\{b_{ijk}^{\alpha\beta}\} \quad \text{y} \quad \vee_k^{\alpha\beta} = \max_{i,j\in[r]}\{b_{ijk}^{\alpha\beta}\}, \quad k \in n.$$

Llamando $I_k^{\alpha\beta}$ al intervalo real cerrado $[\wedge_k^{\alpha\beta}, \vee_k^{\alpha\beta}]$, i.e.

$$I_k^{\alpha\beta} = \{x \in \mathbb{R} \,|\, \wedge_k^{\alpha\beta} \leq x \leq \vee_k^{\alpha\beta}\}. \quad (3)$$

Servirá como el dominio de una función de agregación basada en la desviación que se aplicará como la agregación de $\mathbb{B}_k^{\alpha\beta}$ elementos.

Paso #3 (Implementación de la agregación basada en desviaciones):

Theorem II.2. *Sea I un intervalo de números reales. Sea $\mathbb{X}$ una matriz $s \times t$ tal que $x_{ij} \in I$ por cada $i \in [s]$, $j \in [t]$. Denotamos por $I_{\mathbb{X}}$ un intervalo real cerrado*

$$\big[\min_{i\in[s],j\in[t]}\{x_{ij}\}, \max_{i\in[s],j\in[t]}\{x_{ij}\}\big].$$

Sea $D\colon I_{\mathbb{X}} \times I_{\mathbb{X}} \to \mathbb{R}$ una función de desviación moderada. Se define una función $M_D\colon I_{\mathbb{X}}^{s\cdot t} \to \mathbb{R}$ por

$$M_D(\mathbb{X}) = \frac{1}{2}\big(\sup\{y \in I \mid \sum_{i=1}^{s}\sum_{j=1}^{t} D(x_{ij}, y) < 0\} + \inf\{y \in I \mid \sum_{i=1}^{s}\sum_{j=1}^{t} D(x_{ij}, y) > 0\}\big). \quad \text{(MDX)}$$

entonces M_D es una función de agregación idempotente y simétrica.

Paso #4 (Ponderando las matrices): Sea I un intervalo de números reales. Sea $D\colon I^2 \to \mathbb{R}$ una función de desviación moderada. Sea $\mathbb{W}$ una matriz de ponderación no negativa $s \times t$ tal que $w_{ij} \in [0, \infty)$ por cada $i \in [s]$, $j \in [t]$, y $\mathbb{X}$ será una matriz $s \times t$ de números reales tal que $x_{ij} \in I$ por cada $i \in [s]$, $j \in [t]$. Se dice que el mapeo $M_{D,\mathbb{W}}\colon I^{p\cdot q} \to I$ es una *función de agregación matricial basada en la desviación ponderada* si y solo si

$$M_{D,\mathbb{W}}(\mathbb{X}) = \frac{1}{2}\big(\sup\{y \in I \mid \sum_{i=1}^{s}\sum_{j=1}^{t} w_{ij}D(x_{ij}, y) < 0\} + \inf\{y \in I \mid \sum_{i=1}^{s}\sum_{j=1}^{t} w_{ij}D(x_{ij}, y) > 0\}\big). \quad \text{(MDWX)}$$

La imagen $M_{D,\mathbb{W}}(\mathbb{X})$ de $\mathbb{X}$ se denomina *agregación de matrices basada en desviaciones moderadas ponderada.*

Proposition II.3. *Siendo $k \in [n]$ un número entero escogido arbitrariamente. Sea $\mathbb{B}_k^{\alpha\beta}$ una matriz $r \times r$ definida como Eq. (2). Siendo $\mathbf{w} = (w_1, w_2, \ldots, w_n) \in [0, \infty)^n$ un vector de pesos no negativos. Siendo $M_D : (I_k^{\alpha\beta})^{r\cdot r} \to \mathbb{R}$ una agregación de matrices basada en desviaciones moderadas. Entonces, para cada $k \in [n]$, la función $M_{D,\mathbf{w}}\colon (I_k^{\alpha\beta})^{r\cdot r} \to \mathbb{R}$ definida como:*

$$M_{D,w_k}(\mathbb{B}_k^{\alpha\beta}) = M_D(w_k\mathbb{B}_k^{\alpha\beta}) = M_D\begin{pmatrix} w_k \cdot b_{11k}^{\alpha\beta} & \cdots & w_k \cdot b_{1rk}^{\alpha\beta} \\ w_k \cdot b_{21k}^{\alpha\beta} & \cdots & w_k \cdot b_{2rk}^{\alpha\beta} \\ \vdots & \ddots & \vdots \\ w_k \cdot b_{r1k}^{\alpha\beta} & \cdots & w_k \cdot b_{rrk}^{\alpha\beta} \end{pmatrix} \quad \text{(MDw)}$$

es una agregación de matrices basada en desviaciones moderadas ponderada.

Example II.4. Siendo I un intervalo de números reales cerrado. Siendo $\mathbb{A}$ una matriz de dimensión $12\,000 \times 800$ de cuádruplas reales a_{ij} de modo que

$$a_{ij} = (a_{ij1},\, a_{ij2}, a_{ij3}, a_{ij4}),\ i \in [10\,000],\ j \in [800],$$

y

$$(a_{ij1},\, a_{ij2}, a_{ij3}, a_{ij4}) \in I^5.$$

Siendo $r = 2$. Entonces, siguiendo (1), $\alpha \in [10\,000/2] = [5\,000]$ y $\beta \in [800/2] = [400]$.

Para $\varepsilon \geq 1$, se define $D_\varepsilon\colon I^2 \to \mathbb{R}$ como

$$D_\varepsilon(x, y) = (x + \varepsilon)(y - x).$$

Pues, D_ε es una función de desviación, y la correspondiente función de agregación de dos variables $M_{D_\varepsilon} : I^2 \to \mathbb{R}$ es dada por:

$$M_{D_\varepsilon}(u, v) = \frac{u(u + \varepsilon) + v(v + \varepsilon)}{u + v + 2\varepsilon} \quad (4)$$

para todo u, $v \in I$ siendo $u + v + 2\varepsilon \neq 0$.

Siendo $\mathbf{w} = (w_1, w_2, w_3, w_4) \in [0, \infty)$ un vector de pesos. Para todo $k \in [4]$, $\alpha \in [5\,000]$, $\beta \in [400]$, se define, de

acuerdo con Eq. ((2)), una matriz 2×2 $\mathbb{B}_k^{\alpha\beta}$. Entonces la función $M_{D_\varepsilon,\mathbf{w}}\colon\ I^4 \to \mathbb{R}$:

$$\begin{aligned} M_{D_\varepsilon,w_k}(\mathbb{B}_k^{\alpha\beta}) &= M_{D_\varepsilon}(w_k\mathbb{B}_k^{\alpha\beta}) \\ &= M_{D_\varepsilon}\begin{pmatrix} w_k b_{11k}^{\alpha\beta} & w_k b_{12k}^{\alpha\beta} \\ w_k b_{21k}^{\alpha\beta} & w_k b_{22k}^{\alpha\beta} \end{pmatrix} \\ &= \frac{\sum_{i=1}^{2}\sum_{j=1}^{2} w_k b_{ijk}^{\alpha\beta}(b_{ijk}^{\alpha\beta}+\varepsilon)}{4w_k\varepsilon + \sum_{i=1}^{2}\sum_{j=1}^{2} w_k b_{ijk}^{\alpha\beta}} \end{aligned} \tag{5}$$

es la agregación de matrices basada en desviaciones moderadas ponderada. Finalmente, la resultante matriz de $5\,000 \times 400$ $\mathbb{C}$ de $5\,000 \cdot 400 = 2\,000\,000$ cuadrúplas

$$\mathbf{y}_{\mathbf{w}}^{\alpha\beta} = \big(M_{D_\varepsilon,w_1}(\mathbb{B}_1^{\alpha\beta}), \ldots, M_{D_\varepsilon,w_4}(\mathbb{B}_4^{\alpha\beta})\big),$$

$\alpha \in [5\,000]$, $\beta \in [400]$, es el resultado $\mathbb{A}$ con respecto a D_ε.

B. Enhanced-Multimodal Fusion BCI Framework

El EMF es el marco MI BCI que clasifica señales EEG [25]. El EMF consta de 5 fases diferentes:

1) Calcular la transformada rápida de Fourier de la señal de EEG para transformar los datos de potencia de EEG en el dominio de la frecuencia. Luego, se realiza una diferenciación de la salida FFt en cada frecuencia.
2) Se dividen los datos en cinco bandas de ondas diferentes: δ $1-4$ Hz, θ $4-8$ Hz, α $8-14$ Hz, $\beta 14-30$ Hz y $1-30$ Hz (Todas).
3) Se calcula el CSP en cada banda de onda para extraer características con separación espacial máxima [30].
4) Se entrena un conjunto de clasificadores para cada banda de onda: un análisis discriminante lineal (LDA), análisis discriminante cuadrático (QDA), máquina de vectores de soporte (SVM), K-vecinos cercanos (KNN) y proceso gaussiano (GP). De esta manera tenemos un clasificador de cada tipo para cada banda de onda estudiada.
5) Se realiza la decisión multimodal utilizando dos funciones de agregación. Los mejores resultados en [25] se obtuvieron utilizando una integral de Choquet / Sugeno en la primera fase de la agregación y una función de overlap generalizada [31].

Se puede encontrar un esquema visual del EMF en la Figura 1.

C. Aprendizaje de ponderaciones en una agregación basada en desviación moderada ponderada

Como se detalla en la Sección II-A, es posible usar una MD para agregar un vector de entradas, dando más importancia a algunos canales. Sin embargo, este proceso de ponderación no es sencillo, ya que qué canales deberían ser más importantes que otros depende en gran medida de la tarea a resolver. Además, en el caso de la decisión multimodal, los pesos aprendidos para la fase 1 pueden influir mucho en los pesos óptimos para la fase 2.

En el caso de una tarea de aprendizaje supervisado, podemos plantear este problema como uno de optimización. Usando los logits de los clasificadores como datos de entrenamiento, los agregamos usando la MD ponderada y luego clasificamos cada muestra. La función de coste para la función de optimización es la entropía cruzada para las clases C y las muestras de M:

$$-\sum_{m=1}^{M}\sum_{c=1}^{C} y_{c,m} * log(p_{c,m}) \tag{6}$$

donde y_c, m es un valor binario que es 1 solo si c es la etiqueta real para la muestra m, y $p_{c,m}$ es la probabilidad predicha para la muestra m a ser de clase c.

Procedemos entonces a solucionar este problema mediante la autodiferenciación. El algoritmo de backpropagation para resolver este problema está presente en todas las bibliotecas de cálculo de tensores, que se utilizan comúnmente para Deep Learning [32].

III. Experimentos y resultados

Para nuestros experimentos, hemos utilizado el conjunto de datos BCI Competition IV 2a [33]. Este conjunto de datos consta de cuatro clases de tareas: lengua, pie, mano izquierda y mano derecha realizadas por 9 voluntarios. Para cada tarea, se recolectaron 22 canales de EEG. Hay un total de 288 ensayos para cada participante, distribuidos equitativamente entre las 4 clases. Para nuestra configuración experimental, hemos utilizado 4 de los 22 canales disponibles (8, 12, 14, 18), siguiendo los procedimientos de [24], [25].

De cada sujeto, hemos generado veinte particiones de las 288 pruebas que constan de 50% entrenamiento (144 pruebas) y 50% test (144 pruebas) elegidas al azar. Dado que tenemos 9 sujetos, esto produce un total de 90 conjuntos de datos diferentes.

Estudiamos tanto la clasificación binaria de mano izquierda/derecha como la tarea completa de cuatro clases.

A. Resultados de la clasificación binaria

En esta sección hemos estudiado el rendimiento del EMF utilizando una MD ponderada para la clasificación binaria. Hemos estudiado dos versiones de esta MD ponderada, estableciendo todos los pesos en 1 y aprendiéndolos usando la propagación hacia atrás.

En la Tabla I mostramos los resultados de la MD ponderada en ambos casos y los comparamos con otras agregaciones. Obtuvimos un mejor resultado aprendiendo los pesos que fijándolos al mismo valor, lo que mostró que el algoritmo de entrenamiento funcionó bien. Funcionó de manera similar a la media aritmética y la integral de Choquet, y mejor que la integral de Sugeno. Creemos que esta ventaja de rendimiento se debe a la flexibilidad que se obtiene al utilizar diferentes funciones de agregación en ambas fases.

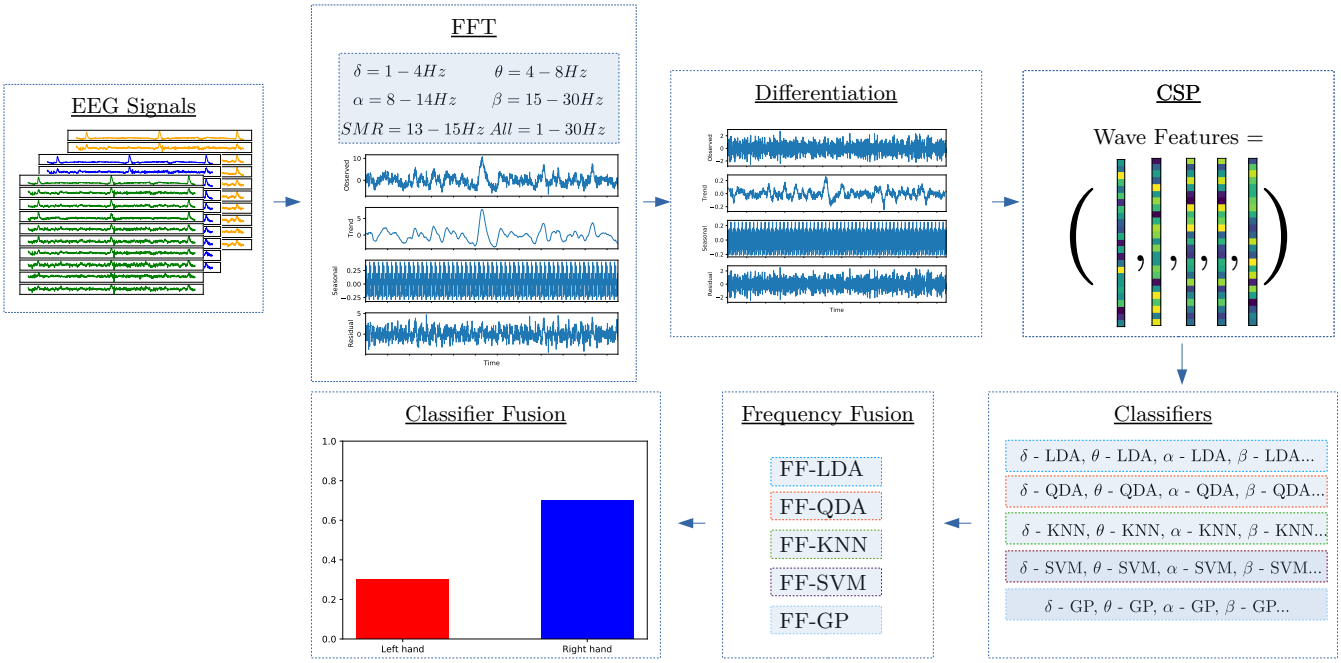


Fig. 1. Esquema visual para el marco BCI de fusión multimodal mejorado.

TABLE I
RESULTADOS DE LA CLASIFICACIÓN BINARIA UTILIZANDO EL EMF CON DIFERENTES CONFIGURACIONES.

Agregación	Tasa de acierto
MD-Ponderada aprendida	$86.97\% \pm 3.98$
MD-Ponderada fijada	$85.80\% \pm 4.08$
Media Aritmética	$85.80\% \pm 4.04$
Integral de Choquet	$86.39\% \pm 4.20$
Integral de Sugeno	$81.39\% \pm 4.39$

TABLE II
RESULTADOS DE LA CLASIFICACIÓN DE LAS CUATRO CLASES UTILIZANDO EL EMF CON DIFERENTES CONFIGURACIONES.

Agregación	Tasa de acierto
MD-Ponderada aprendida	$65.91\% \pm 13.15$
MD-Ponderada fijada	$72.93\% \pm 2.29$
Media Aritmética	$72.22\% \pm 2.31$
Integral de Choquet	$72.93\% \pm 1.85$
Integral de Sugeno	$64.45\% \pm 2.66$

B. Resultados de la tarea completa

En esta sección, hemos estudiado el rendimiento del sistema BCI EMF utilizando la MD ponderada para la tarea de clasificación de cuatro clases. Hemos estudiado la versión de pesos fijos de la MD, donde todos los canales son igualmente importantes, y la versión de peso aprendido usando la propagación hacia atrás, al igual que en la tarea de clasificación binaria.

Hemos mostrado nuestros resultados en la Tabla II. En esta Tabla también encontramos los resultados para la media aritmética y las integrales de Sugeno y Choquet. Descubrimos que la MD ponderada funciona igual o mejor que las otras integrales. Aprender el peso del canal también nos dio un peor resultado que fijar todos los pesos a 1, probablemente por la falta de suficientes datos. El mejor resultado se obtuvo usando la MD ponderado fija y la integral de Choquet, que resultaron en una precisión de $72,93\%$.

IV. CONCLUSIONES

En este trabajo hemos presentado las agregaciones ponderadas basadas en desviaciones moderadas aplicadas a un marco de interfaz cerebro-computadora de imágenes motoras, el marco de fusión multimodal mejorado. Hemos mostrado en qué parte del marco BCI se aplica y cómo se pueden optimizar los pesos de esta función mediante la diferenciación automática.

Hemos comparado los resultados utilizando pesos optimizados y fijos, obteniendo mejores resultados al aprenderlos. Asimismo, hemos comparado las agregaciones basadas en Desviación Moderada con el promedio aritmético y las integrales de Sugeno y Choquet, obteniendo resultados favorables a nuestras soluciones.

Nuestra investigación futura tendrá como objetivo mejorar el proceso de aprendizaje para requerir menos muestras de entrenamiento y mejorar aún más la fase de salida del clasificador explorando más funciones de desviación moderada.

AGRADECIMIENTOS

La investigación de Javier Fumanal Idocin, A.Urio y Humberto Bustince ha sido apoyada por el proyecto PID2019-108392GB I00 (AEI / 10.13039 / 501100011033).

La investigación de Carmen Vidaurre ha sido financiada por el proyecto RyC-2014-15671.

REFERENCES

[1] C. Vidaurre, C. Klauer, T. Schauer, A. Ramos-Murguialday, and K.-R. Müller, "Eeg-based bci for the linear control of an upper-limb neuroprosthesis," *Medical engineering & physics*, vol. 38, pp. 1195–1204, 2016.

[2] B. Blankertz, L. Acqualagna, S. Dähne, S. Haufe, M. Schultze-Kraft, I. Sturm, M. Uscumlic, M. Wenzel, G. Curio, and K. Müller, "The berlin brain-computer interface: Progress beyond communication and control," *Front Neurosci*, vol. 10, p. 530, 2016.

[3] B. Blankertz, S. Lemm, M. Treder, S. Haufe, and K.-R. Müller, "Single-trial analysis and classification of ERP components–a tutorial," *NeuroImage*, vol. 56, no. 2, pp. 814–825, 2011.

[4] C. Sannelli, C. Vidaurre, K. Müller, and B. Blankertz, "A large scale screening study with a smr-based bci: Categorization of bci users and differences in their smr activity," *PLOS ONE*, vol. 14, p. e0207351, 2019.

[5] R. Scherer and C. Vidaurre, "Chapter 8 - motor imagery based brain-computer interfaces," in *Smart Wheelchairs and Brain-Computer Interfaces*, P. Diez, Ed. Academic Press, 2018, pp. 171–195.

[6] B. J. Lance, J. Touryan, Y.-K. Wang, S.-W. Lu, C.-H. Chuang, P. Khooshabeh, P. Sajda, A. Marathe, T.-P. Jung, C.-T. Lin *et al.*, "Towards serious games for improved bci," *Handbook of Digital Games and Entertainment Technologies. Singapore: Springer*, pp. 1–28, 2015.

[7] C. Vidaurre, S. Haufe, T. Jorajurí´a, K.-R. Müller, and V. V. Nikulin, "Sensorimotor functional connectivity: A neurophysiological factor related to bci performance," *Frontiers in Neuroscience*, vol. 14, p. 1278, 2020.

[8] C. Vidaurre, R. Scherer, R. Cabeza, A. Schlögl, and G. Pfurtscheller, "Study of discriminant analysis applied to motor imagery bipolar data," *Medical & biological engineering & computing*, vol. 45, no. 1, p. 61, 2007.

[9] Y. R. Tabar and U. Halici, "A novel deep learning approach for classification of eeg motor imagery signals," *Journal of neural engineering*, vol. 14, no. 1, p. 016003, 2016.

[10] Y.-K. Wang, T.-P. Jung, and C.-T. Lin, "Eeg-based attention tracking during distracted driving," *IEEE transactions on neural systems and rehabilitation engineering*, vol. 23, no. 6, pp. 1085–1094, 2015.

[11] Y. Zhang, C. S. Nam, G. Zhou, J. Jin, X. Wang, and A. Cichocki, "Temporally constrained sparse group spatial patterns for motor imagery bci," *IEEE Transactions on Cybernetics*, vol. 49, no. 9, pp. 3322–3332, Sep. 2019.

[12] B. Blankertz, R. Tomioka, S. Lemm, M. Kawanabe, and K.-R. Muller, "Optimizing spatial filters for robust eeg single-trial analysis," *IEEE Signal processing magazine*, vol. 25, no. 1, pp. 41–56, 2008.

[13] C. Sannelli, C. Vidaurre, K.-R. Müller, and B. Blankertz, "CSP patches: an ensemble of optimized spatial filters. an evaluation study," *Journal of Neural Engineering*, vol. 8, no. 2, p. 025012, 2011.

[14] K.-R. Müller, C. W. Anderson, and G. E. Birch, "Linear and Non-Linear Methods for Brain-Computer Interfaces," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 11, no. 2, pp. 165–169, 2003.

[15] T. Nierhaus, C. Vidaurre, C. Sannelli, K.-R. Mueller, and A. Villringer, "Immediate brain plasticity after one hour of brain–computer interface (bci)," *The Journal of Physiology*, vol. n/a, no. n/a, 2019.

[16] C. Vidaurre, A. R. Murguialday, S. Haufe, M. Gómez, K.-R. Müller, and V. Nikulin, "Enhancing sensorimotor bci performance with assistive afferent activity: An online evaluation," *NeuroImage*, vol. 199, pp. 375 – 386, 2019.

[17] M. Mironovova and J. Bíla, "Fast fourier transform for feature extraction and neural network for classification of electrocardiogram signals," in *2015 Fourth International Conference on Future Generation Communication Technology (FGCT)*, 2015, pp. 1–6.

[18] W. Zheng, W. Liu, Y. Lu, B. Lu, and A. Cichocki, "Emotionmeter: A multimodal framework for recognizing human emotions," *IEEE Transactions on Cybernetics*, vol. 49, no. 3, pp. 1110–1122, Mar. 2019.

[19] M. Murugappan, S. Murugappan, C. Gerard *et al.*, "Wireless eeg signals based neuromarketing system using fast fourier transform (fft)," in *2014 IEEE 10th International Colloquium on Signal Processing and its Applications*. IEEE, 2014, pp. 25–30.

[20] D. Iacoviello, A. Petracca, M. Spezialetti, and G. Placidi, "A classification algorithm for electroencephalography signals by self-induced emotional stimuli," *IEEE Transactions on Cybernetics*, vol. 46, no. 12, pp. 3171–3180, Dec. 2016.

[21] L. Xie, Z. Deng, P. Xu, K. Choi, and S. Wang, "Generalized hidden-mapping transductive transfer learning for recognition of epileptic electroencephalogram signals," *IEEE Transactions on Cybernetics*, vol. 49, no. 6, pp. 2200–2214, Jun. 2019.

[22] S. Bhattacharyya, A. Khasnobish, S. Chatterjee, A. Konar, and D. N. Tibarewala, "Performance analysis of lda, qda and knn algorithms in left-right limb movement classification from eeg data," in *2010 International Conference on Systems in Medicine and Biology*, Dec. 2010, pp. 126–131.

[23] G. Dornhege, B. Blankertz, G. Curio, and K. . Muller, "Boosting bit rates in noninvasive eeg single-trial classifications by feature combination and multiclass paradigms," *IEEE Transactions on Biomedical Engineering*, vol. 51, no. 6, pp. 993–1002, 2004.

[24] L.-W. Ko, Y.-C. Lu, H. Bustince, Y.-C. Chang, Y. Chang, J. Ferandez, Y.-K. Wang, J. A. Sanz, G. P. Dimuro, and C.-T. Lin, "Multimodal fuzzy fusion for enhancing the motor-imagery-based brain computer interface," *IEEE Computational Intelligence Magazine*, vol. 14, no. 1, pp. 96–106, 2019.

[25] J. Fumanal-Idocin, Y.-K. Wang, C.-T. Lin, J. Fernández, J. Sanz, and H. Bustince, "Motor-imagery-based brain computer interface using signal derivation and aggregation functions," *arXiv preprint arXiv:2101.06968*, 2021.

[26] A. H. Altalhi, J. I. Forcén, M. Pagola, E. Barrenechea, H. Bustince, and Z. Takac, "Moderate deviation and restricted equivalence functions for measuring similarity between data," *Information Sciences*, vol. 501, pp. 19–29, 2019.

[27] M. Papčo, I. Rodríguez-Martínez, J. Fumanal-Idocin, A. H., and H. Bustince, "A fusion method for multi-valued data," *Information Fusion*, vol. 71, pp. 1–10, 2021.

[28] G. Beliakov, H. Bustince, and T. C. Sánchez, *A practical guide to averaging functions*. Springer, 2016, vol. 329.

[29] G. Beliakov, A. Pradera, T. Calvo *et al.*, *Aggregation functions: A guide for practitioners*. Springer, 2007, vol. 221.

[30] Y. Wang, S. Gao, and X. Gao, "Common spatial pattern method for channel selelction in motor imagery based brain-computer interface," in *2005 IEEE Engineering in Medicine and Biology 27th Annual Conference*, 2005, pp. 5392–5395.

[31] L. D. Miguel, D. Gómez, J. T. Rodríguez, J. Montero, H. Bustince, G. P. Dimuro, and J. A. Sanz, "General overlap functions," *Fuzzy Sets and Systems*, vol. 372, pp. 81–96, 2019, theme: Aggregation Operations.

[32] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," 2017.

[33] M. Tangermann, K.-R. Müller, A. Aertsen, N. Birbaumer, C. Braun, C. Brunner, R. Leeb, C. Mehring, K. J. Miller, G. Mueller-Putz *et al.*, "Review of the bci competition iv," *Frontiers in neuroscience*, vol. 6, p. 55, 2012.

Operador de comparación de elementos multivaluados basado en Funciones de Equivalencia Restringida

1º Aitor Castillo-López[a], 2º Carlos Lopez-Molina[a,b,c], 3º Javier Fernández[a], 4º Humberto Bustince[a] y 5º Sesma-Sara[a]
[a]*Departamento de Estadística, Informática y Matemática, Universidad Pública de Navarra*, (31006) Pamplona, Spain
[b]*KERMIT, Dept. Data Analysis and Mathematical Modelling, Ghent University, (9000) Ghent, Belgium*
[c]*NavarraBiomed, Complejo Hospitalario de Navarra, (31008) Pamplona, Spain*
aitor.castillo@unavarra.es

***Abstract*—En este trabajo proponemos un nuevo enfoque del algoritmo de clustering gravitacional basado en lo que Einstein consideró su "mayor error": la constante cosmológica. De manera similar al algoritmo de clustering gravitacional, nuestro enfoque está inspirado en principios y leyes del cosmos, y al igual que ocurre con la teoría de la relatividad de Einstein y la teoría de la gravedad de Newton, nuestro enfoque puede considerarse una generalización del agrupamiento gravitacional, donde, el algoritmo de clustering gravitacional se recupera como caso límite. Además, se desarrollan e implementan algunas mejoras que tienen como objetivo optimizar la cantidad de iteraciones finales, y de esta forma, se reduce el tiempo de ejecución tanto para el algoritmo original como para nuestra versión.**

***Index Terms*—cosmos, clustering, no supervisado, simulación, fuerza gravitacional**

I. Introducción

LOS OPERADORES DE COMPARACIÓN han sido objeto de estudio en el ámbito del procesamiento de la información. De hecho, la comparación (cuantitativa) de información es una de las tres operaciones más básicas sobre datos, junto a las operaciones de igualdad y ordenación. De manera general, la investigación se ha centrado en simular el comportamiento humano al realizar este tipo de operaciones.

Gran parte de la literatura sobre operadores de comparación está dedicada a las métricas, así como a clases de operadores estrechamente relacionados (pseudométricas [1], cuasimétricas [2], etc.). Una de las principales críticas a las métricas como operadores de comparación es el hecho de que se imponga la desigualdad triangular [3]. Si bien la desigualdad triangular es matemáticamente conveniente en una amplia gama de escenarios, no está claro si los humanos realmente se comportan de acuerdo con esta propiedad, y se pueden encontrar muchos contraejemplos diferentes en contextos específicos [4]. Por esta razón, los investigadores han intentado construir paradigmas de comparación que no se basen ni se inspiren en métricas.

Dentro de la teoría de conjuntos difusos, la comparación se ha abordado de diferentes maneras. Una parte importante se ha dedicado a la idea de métricas difusas [5], [6] o pseudométricas [7], [8], y las Funciones de Equivalencia Restringida (REF por sus siglas en inglés), son en este ámbito de vital relevancia. Estas fueron presentadas en [9] para la comparación de grados de pertenencia en el intervalo $[0, 1]$ adaptando los axiomas originales propuestos por J. Fodor y M. Roubens [10]. Desde su introducción, el concepto de REF se ha adaptado a ámbitos en los que los valores a comparar se encuentran dentro del intervalo $[0, 1]$. Ejemplos relevantes son las REF intervalo-valoradas (IV-REF), diseñadas para comparar grados de pertenencia con intervalos [11], o las REF radiales (RREF), adaptadas para datos escalares en configuraciones radiales [12]. Una necesidad crítica en la adaptación de las REF a escenarios distintos al original es en relación al modelado del orden de crecimiento, que se utiliza críticamente en la definición axiomática de REF.

En este trabajo presentamos una adaptación de las REF a datos multivaluados, que denotamos como L^n. Para lograr este objetivo, presentamos la idea de L^n-REF, y desarrollamos un conjunto de axiomas que estos operadores deben cumplir. Además, introducimos métodos de construcción para L^n-REF capaces de acomodar diferentes interpretaciones en el ordenamiento multivaluado. Nuestras propuestas, en términos de ordenación de datos multivaluados, se cicunscriben dentro del la taxonomía de Barnett [13]. Tengase en cuenta que imponer algún orden para los datos multivaluados es necesariamente arbitrario y dependiente del contexto, ya que *no hay un orden natural para los datos multivaluados* [13]. A modo de ejemplo, utilizamos las L^n-REF en la comparación de colores, ya que el color se representa de forma habitual como un dato multivaluado (independientemente del espacio de color específico). Además sirve como ejemplo de aplicación de las L^n-REF en visión artificial.

El resto de este trabajo está organizado de la siguiente manera. La Sección II resume algunos conceptos de uso para las próximas secciones. La Sección III presenta nuestra propuesta, mientras que diferentes ejemplos ilustrativos presentan una prueba de concepto de nuestros operadores en el contexto de la comparación de colores. Finalmente, en la Sección IV se recogen las conclusiones de nuestro trabajo y posibles líneas futuras.

II. Preliminares

A. REF en L

Considérese $(L, \leq)$ donde $L = [0,1]$ y $\leq$ es el orden natural de los números. Ahora bien, considérense en letra mayúscula los elementos pertenecientes a L^n, es decir $X = (x_1, \ldots, x_n) \in L^n$ donde $n \in \mathbf{N}$. Existe un orden parcial $\leq_P$ inducido por $\leq$ dado de la siguiente manera: $X \leq_P Y$ iff $x_i \leq y_i$ para todo $i \in \{1, \ldots, n\}$[1].

Denótense $\mathbf{0} = (0, \ldots, 0) \in L^n$ y $\mathbf{1} = (1, \ldots, 1) \in L^n$. Una función de agregación n-aria M de elementos multivaluados en L^n es una función $M : L^n \to L$ creciente para cada una de las variables y debe satisfacer $M(\mathbf{0}) = 0$, $M(\mathbf{1}) = 1$ [14]–[16]. Las siguientes propiedades para las funciones de agregación $M : L^n \to L$ son de utilidad para las siguientes secciones:

(P1) $M(x_1, \ldots, x_n) = 0$ iff $x_1 = \ldots = x_n = 0$.
(P2) $M(x_1, \ldots, x_n) = 1$ iff $x_1 = \ldots = x_n = 1$.

Una Media Aritmética Ponderada n-aria (MAP) en L con pesos normalizados $w_1, \ldots, w_n \in L$ y $X = (x_1, \ldots, x_n)$ es una función $\omega : L^n \to L$ definida como $\omega(X) = w_1 x_1 + \ldots + w_n x_n$ tal que $w_1 + \ldots + w_n = 1$.

Definición II.1. *Un automorfismo de L es una función continua estrictamente creciente $\varphi : L \to L$ tal que $\varphi(0) = 0$ y $\varphi(1) = 1$. Además, la identidad en L se indica con Id.*

Defínase una REF en L construida por automorfismos de la siguiente manera.

Definición II.2. *[9] Una función $R : [0,1]^2 \to [0,1]$ es llamada Función de Equivalencia Restringida si cumple:*

(R1) $R(x, y) = 1$ *iff* $x = y$;
(R2) $R(x, y) = 0$ *iff* $\{x, y\} = \{0, 1\}$;
(R3) $R(x, y) = R(y, x)$ *para todo* $x, y \in [0, 1]$;
(R4) *Si* $x \leq y \leq z$, *entonces* $R(x, z) \leq R(x, y)$ *y* $R(x, z) \leq R(y, z)$ *para todo* $x, y, z \in [0, 1]$.

En [17] se introduce un método para construir REFs en términos de automorfismos.

Proposición II.3. *[17] Si φ_1, φ_2 son dos automorfismos de L, entonces la funcion $R : L^2 \to L$ definida como*

$$R(x, y) = \varphi_1^{-1}\Big(1 - |\varphi_2(x) - \varphi_2(y)|\Big) ,$$

es una REF.

Definición II.4. *Una función $f : (L^n)^m \to L^n$ es llamada representable si existen $f_1, \ldots, f_n : L^m \to L$ tales que*

$$f(X_1, \ldots, X_m) = \big(f_1(x_{11}, \ldots, x_{m1}), \ldots, f_n(x_{1n}, \ldots, x_{mn})\big) , \quad (1)$$

para todo $X_1, \ldots, X_m \in L^n$ with $X_i = (x_{i1}, \ldots, x_{in})$ para todo $i \in \{1, \ldots, m\}$.

[1] A lo largo de este trabajo, la definición de i será la misma para abreviar. De lo contrario, se redefinirá explícitamente para algunas excepciones si así fuera necesario.

B. Espacios de color y operadores lineales

Como se expone en [18], para reproducir una imagen a color, es necesario generar nuevos vectores en el espacio espectral n a partir de los obtenidos por un sensor multiespectral dado, mientras que los dispositivos de salida, que pueden caracterizarse como aditivos o sustractivos, debe poder reproducir colores de estos vectores. Dado que el ojo humano se puede representar como un sensor $n = 3$, lo más común es usar espacios tridimensionales para guardar información de color. Entre todos los diferentes sistemas de color posibles para dispositivos de salida aditiva, el modelo RGB [19] es el más extendido debido a la evolución de la codificación, que trata las imágenes a color como tres bandas monocromáticas independientes [20]. A pesar de que existen otros espacios de color como *CIE* $L^*a^*b^*$ que intentan reproducir un espacio de color uniforme basado en la percepción humana del color, donde el significado de cada valor L^*, a^* y b^* es bastante diferente. Por el contrario, el espacio RGB es un espacio simple en forma de cubo y el significado de cada valor que compone un *triplete* es siempre el mismo, siendo la cantidad de rojo, verde o azul que compone un color respectivamente. Por simplicidad e idoneidad, tomaremos el modelo RGB como espacio de color en el que trabajar. Incluso si se puede elegir cualquier otro espacio de color, los resultados y las interpretaciones podrían ser bastante diferentes.

En RGB es habitual representar los tres componentes de un color en una escala de 0 a 255 y sólo se corresponde un único color para cada triplete en el espacio $[0, 255]^3$. Consideremos ahora el espacio RGB, pero refactorizado al espacio $[0, 1]^3$, es decir, L^3. Entonces, los colores que corresponden a las esquinas del cubo RGB son:

- Rojo: C_R $(1, 0, 0)$
- Verde: C_G $(0, 1, 0)$
- Azul: C_B $(0, 0, 1)$
- Cian: C_C $(0, 1, 1)$
- Magenta:C_M $(1, 0, 1)$
- Amarillo: C_Y $(1, 1, 0)$
- Negro: C_K $(0, 0, 0)$ o $\mathbf{0}$
- Blanco: C_W $(1, 1, 1)$ o $\mathbf{1}$

Estos son los únicos colores compuestos usando solo 0s y 1s y por esta razón los llamamos *tripletes crisp* o *colores crisp*.

Todos los colores posibles en el espacio RGB se pueden describir, de acuerdo con la terminología común, en los siguientes términos de apariencia de color [21]: *brillo*, *tonalidad o matiz* y *saturación*. Un color pierde brillo si disminuye el promedio de los valores del triplete que describe su posición en el espacio RGB. Entonces, todos los colores dentro de un plano perpendicular a la diagonal entre $\mathbf{0}$ y $\mathbf{1}$ tienen brillo idéntico. El tono del color es otra propiedad principal y está relacionada con la longitud de onda dominante percibida; en otras palabras, si un color es rojo, azul, amarillo, naranja... Los colores dentro de la diagonal entre $\mathbf{0}$ y $\mathbf{1}$ son los únicos colores sin matiz, y además, en esta diagonal se encuentran todos los colores grises posibles; por eso dicha recta se llama *diagonal de grises*. Finalmente, la saturación describe cuán cerca está

un triplete de la diagonal de grises; cuanto más cerca(lejos) de la diagonal de grises, menor(mayor) saturación. A mayor saturación, más fácil es percibir el tono de un color dado.

Denotamos *colores complementarios* a los pares de colores que, cuando se combinan, producen un determinado color en escala de grises [22]. Estos pares de colores se consideran complementarios dependiendo de la teoría de color que se utilice: la teoría de color moderna utiliza el modelo de color aditivo RGB o el modelo CMY para los sustractivos. Es por ello que en este trabajo consideramos *colores complementarios crisp* los pares $C_R - C_C$, $C_G - C_M$, y $C_B - C_Y$. El par de colores $C_K - C_W$ es común a todas las teorías del color.

A lo largo de todo este trabajo, se pueden aplicar diferentes $f : (L^n)^m \to L^n$ (como se presenta en Def. II.4) a un punto $X = (x_1, \ldots, x_n) \in L^n$, es decir, $f(X \times m)$. Además, algunas funciones f se construyen con $m = n$ MAPs. En estos casos, la función f toma la forma

$$f(X \times n) = \big(\omega_1(X), \ldots, \omega_n(X)\big) \ ,$$

que se puede reescribir como una transformación lineal de un punto $X \in L^n$ a otro punto $X' \in L^n$ mediante una multiplicación matricial;

$$\begin{aligned} \big(f(X \times n)\big)^T = \\ \begin{pmatrix} \omega_1(X) \\ \vdots \\ \omega_n(X) \end{pmatrix} = \begin{pmatrix} w_{11}x_1 + \cdots + w_{1n}x_n \\ \vdots \qquad \vdots \qquad \vdots \\ w_{n1}x_1 + \cdots + w_{nn}x_n \end{pmatrix} = \\ \begin{pmatrix} w_{11} & \cdots & w_{1n} \\ \vdots & & \vdots \\ w_{n1} & \cdots & w_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} = WX^T = X'^T \ , \end{aligned} \tag{2}$$

donde la fila i-ésima de la matriz W se compone de los pesos de la i-ésima MAP, ω_i.

Esta forma de representación nos permite tratar muchas de las futuras operaciones en este trabajo como operaciones lineales algebraicas en el espacio RGB y se utilizarán para:

1) Visualizar de una manera sencilla lo que sucede con el espacio de color, mientras,
2) La notación algebraica reduce la complejidad de las expresiones obtenidas, y además,
3) Simplifica la implementación del algoritmo presentado.

Nótese que para cualquier n el cubo L^n se transforma en un *n-hiperparalelepípedo* más pequeño dentro del cubo original L^n con dos de sus vértices opuestos en $\mathbf{0}$ y $\mathbf{1}$ debido a las propiedades (P1) y (P2). Además, sea $X_d = (x, \ldots, x)$ cualquier punto en la diagonal de L^n, *i.e.*, X_d está en la línea entre $\mathbf{0}$ y $\mathbf{1}$. Es fácil ver que X_d permanecerá inmutable bajo la transformación W presentada en la Eq. 2 debido a que cada una de las filas de W suma 1, entonces,

$$X_d'^T = WX_d^T = X_d^T . \tag{3}$$

En términos de color, la Eq. 3 implica que cualquier color gris es inmutable bajo cualquier W. Con respecto a los colores de fuera de la diagonal de grises, estos serán diferentes bajo una transformación W. De hecho, los colores que no son grises

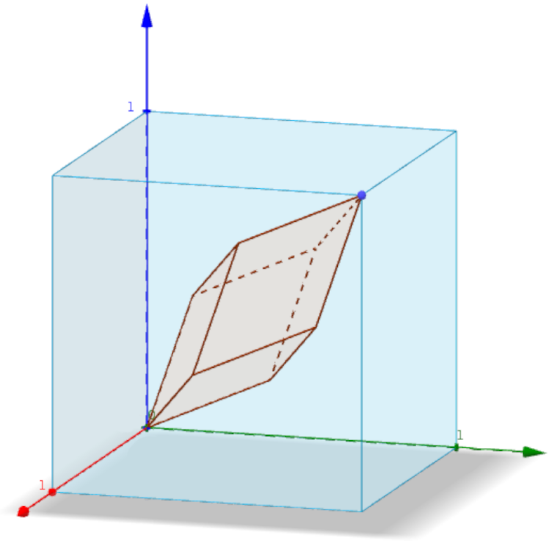


Fig. 1: Transformación del cubo L^3 para los valores del Ejemplo II.5. Es evidente que las propiedades del espacio resultante se pueden deducir de las propiedades de la matriz W como, por ejemplo, sus simetrías, el determinante, etc.

se aproximarán a la diagonal de la escala de grises cuando $det(W) < 1$, por lo que los colores perderán saturación. En muchas otras transformaciones, estos colores también pueden cambiar su tono y brillo.

Ejemplo II.5. *Tomemos la siguiente matriz W para un ejemplo en L^3,*

$$W = \begin{pmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{2} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \end{pmatrix} .$$

Teniendo en cuenta las columnas de W, es fácil ver que los tripletes C_R, C_G y C_B, que son la base de nuestro espacio RGB, se transforman en ■ $= \left(\frac{1}{3}, \frac{1}{2}, \frac{1}{4}\right)$, ■ $= \left(\frac{1}{3}, \frac{1}{4}, \frac{1}{2}\right)$ *y* ■ $= \left(\frac{1}{3}, \frac{1}{4}, \frac{1}{4}\right)$ *respectivamente. La representación visual de la transformación del espacio RGB se muestra en la Figura 1. Como se deduce de la Eq. 3, cualquier color de la escala de grises permanecerá invariante bajo transformación. También se puede apreciar cómo los colores de las esquinas C_R, C_G y C_B han perdido saturación, y además, han cambiado de tono debido a la rotación inducida por W. Los colores C_R y C_G han ganado brillo, mientras C_B lo ha perdido.*

Como resumen, si aplicamos esta W a los colores que forman la base del espacio RGB, C_R, C_G y C_B, se transformarán en

■■■ → ■■■.

III. L^n-REF: Nuevo operador para comparar dos colores obteniendo un tercero como resultado

El objetivo de esta sección es construir un operador que sea capaz de devolver una medida de equivalencia en L^n como resultado de comparar dos elementos multivaluados pertenecientes a dicho espacio. Para ello, mientras la teoría es desarrollada para cualquier n, los ejemplos ilustrativos se darán para $n = 3$, y en concreto, se presentará un algoritmo de comparación de imágenes. Este, dadas dos imágenes, debe devolver una imagen de salida donde sus pixeles pueden ser interpretados como un mapa, donde sea posible interpretar las regiones donde las imágenes de entrada son similares en color o, en su defecto, diferentes. Para lograr este objetivo, introducimos un algoritmo para comparar pixel a pixel todos los pixeles de una imagen con los pixeles correspondientes en una segunda imagen de entrada, siendo necesario que, ambas tengan las mismas dimensiones.

Definamos el pixel ij-ésimo de una imagen A como $P_{A_{ij}}$. En el Algoritmo 1 presentamos la estructura computacional de nuestra propuesta.

Algoritmo 1: Comparación pixel a pixel

Entrada: Dos imagenes A y B del mismo tamaño.
Resultado: Una imagen a color de tamaño idéntico.
Escoger un método de comparación de pixeles ;
para *cada posición* ij **hacer**
  Comparar los pixeles $P_{A_{ij}}$ y $P_{B_{ij}}$;
  Asignar al ij-ésimo pixel de la imagen resultado el valor de la comparación entre pixeles anterior;
fin

Por lo tanto, para la acción de compararación del Alg. 1 es necesario construir un método para comparar pixeles, es decir, encontrar un operador que dados dos tripletes de entrada su salida sea otro triplete que guarde la información de la comparación.

Llegados a este punto, podemos considerar dos filosofías diferentes teniendo en cuenta cuáles de los tripletes crisp complementarios son antagonistas para el caso del color. Es posible construir dos métodos diferentes, uno para cada una de las siguientes filosofías:

1) Un método que trata el par de colores complementarios $C_K - C_W$ de manera diferente a los otros pares crisp complementarios al compararlos, siendo $C_K - C_W$ el único par que presenta el mayor antagonismo.
2) Un método que trata todos los pares crisp complementarios por igual al compararlos.

Para comparar dos pixeles de color, debemos definir la noción de REF sobre L^3 (en general sobre L^n), de tal forma que el valor que obtengamos sea nuevamente un elemento de L^3 (L^n). En las siguiente subsección definimos y presentamos una versión de REF en L^n con este propósito. De las dos filosofías posibles, en este trabajo solamente se ha desarrollado la primera planteada donde el par $C_K - C_W$ se trata de diferente manera que los restantes.

A. L^n-REF *basadas en la filosofía donde el par* $\mathbf{0} - \mathbf{1}$ *es tratado de diferente manera*

Esta versión de REF trata el par $C_K - C_W$ como el único que representa la menor eequivalencia posible.

Definición III.1. *Sea n un número entero positivo. Una función $R_{L^n} : L^n \times L^n \to L^n$ se llama función de equivalencia restringida en L^n (R_{L^n}), si satisface:*

(RL1) $R_{L^n}(X, Y) = \mathbf{1}$ *iff* $X = Y$;
(RL2) $R_{L^n}(X, Y) = \mathbf{0}$ *iff* $\{X, Y\} = \{\mathbf{0}, \mathbf{1}\}$;
(RL3) $R_{L^n}(X, Y) = R_{L^n}(Y, X)$ *para todo* $X, Y \in L^n$;
(RL4) *Si* $X \leq_P Y \leq_P Z$, *entonces* $R_{L^n}(X, Z) \leq_P R_{L^n}(X, Y)$ *y* $R_{L^n}(X, Z) \leq_P R_{L^n}(Y, Z)$ *para todo* $X, Y, Z \in L^n$.

Teniendo en cuenta Def. II.2, la justificación del axioma (RL1) es natural; comparar tripletes equivalentes debe devolver el valor más alto en L^n, *i.e.*, $\mathbf{1}$, como medida de equivalencia dado que estos son equivalentes. En cuanto al axioma (RL2), se trata de comparar los tripletes $\mathbf{0}$ y $\mathbf{1}$. En este caso, estamos teniendo en cuenta que los tripletes C_K y C_W son los tripletes menos equivalentes entre todos los pares de colores posibles según la filosofía escogida, entonces, siendo este nuestro punto de partida, se justifica que son los tripletes únicos que al compararse, deben devolver $\mathbf{0}$, es decir, el valor más bajo posible de todo L^n. La justificación del axioma (RL3) es que se exige a la REF que la comparación entre dos pixeles debe cumplir con la simetría. Esta propiedad puede no ser necesaria en otras aplicaciones donde existe una dependencia entre imágenes comparadas (como imágenes de vídeo donde existe una relación temporal entre fotogramas), en nuestro caso, no se considera dependencia del tiempo, por lo que (RL3) está justificado. Finalmente, la justificación del axioma (RL4 $_1$) es que la equivalencia resultante entre comparar dos tripletes similares debe ser mayor que la equivalencia de comparar dos tripletes que son, al menos, más diferentes que los anteriores.

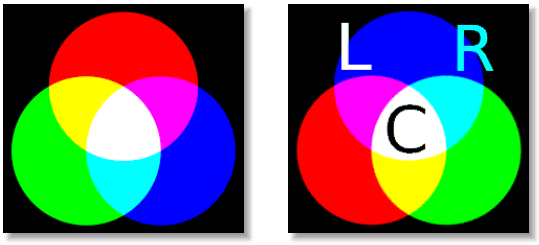
Ahora damos un método de construcción para R_{L^n} en L^n de acuerdo con Def. III.1.

Teorema III.2. *Sea $\omega_i : L^n \to L$ una MAP n-aria como vector de pesos normalizados $(w_{i1}, \ldots, w_{in})$ tal que los vectores son linealmente independientes y existe $k \in \{1, \ldots, n\}$ con $w_{kj} \neq 0$ para todo $j \in \{1, \ldots, n\}$. Sea $\mathbf{R} = (R_1, \ldots, R_n)$ una secuencia de REFs en L. Entonces la función $R_{L^n} : L^n \times L^n \to L^n$ dada por,*

$$R_{L^n}(X, Y) = \left(R_1\Big(\omega_1(X), \omega_1(Y)\Big), \ldots, R_n\Big(\omega_n(X), \omega_n(Y)\Big) \right), \tag{4}$$

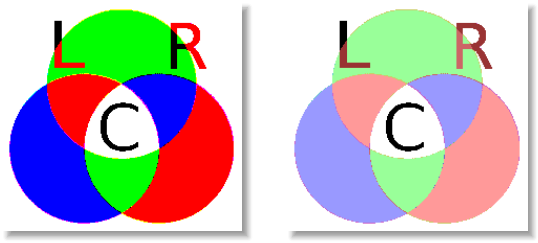
para todo $X, Y \in L^n$, es una R_{L^n} en L^n.

Proof. (RL1) La suficiencia se deriva de Eq. (4). Con respecto a la necesidad, sea $R_{L^n}(X, Y) = \mathbf{1}$, entonces, $R_i\Big(\omega_i(X), \omega_i(Y)\Big) = 1$, por lo tanto $\omega_i(X) = \omega_i(Y)$ y finalmente $X = Y$.



(a) Img. A (b) Img. B

Fig. 2: Las imágenes sintéticas A (Fig. 2a) y B (Fig. 2b) son el input para el Algoritmo 1 en el Ejemplo III.4.



(a) $Alg1(A,B)^{W=\mathbb{I}}_{R_{L_2}}$ (b) $Alg1(A,B)^{W=\Gamma}_{R_{L_2}}$

Fig. 3: Mapas resultantes para $W=\mathbb{I}$ (Fig. 3a), y para $W=\Gamma$ (Fig. 3b).

(RL2) Sea $R_{L^n}(X,Y)=\mathbf{0}$. Por lo tanto para todo i, se tiene que $R_i\Big(\omega_i(X),\omega_i(Y)\Big)=0$, de modo que $\{\omega_i(X),\omega_i(Y)\}=\{0,1\}$. Ya que existe k tal que ω_k satisface (P1) y (P2), se sigue que $\{X,Y\}=\{\mathbf{0},\mathbf{1}\}$.

(RL3) La prueba es directa teniendo en cuenta (R3).

(RL4) De $x_i\leq y_i\leq z_i$ se obtiene $\omega_j(X)\leq\omega_j(Y)\leq\omega_j(Z)$ para todo j y consecuentemente $R_j\left(\omega_j(X),\omega_j(Z)\right)\leq R_j\left(\omega_j(X),\omega_j(Y)\right)$, de donde se sigue que $R_{L^n}(X,Z)\leq_P R_{L^n}(X,Y)$. La prueba para $R_{L^n}(X,Z)\leq_P R_{L^n}(Y,Z)$ es similar. □

Si aplicamos el método de construcción de las REFs $R_1,\ldots,R_n$ en el Teorema. III.2 en términos de automorfismos se obtiene que al seguir la construcción de R_{L^n}.

Corolario III.3. *Si se asume el Teorema III.2, sea φ_{ij} para $j=1,2$, un automorfismo de L. Luego la función $R_{L^n}:L^n\times L^n\to L^n$ dada por,*

$$R_{L^n}(X,Y)=\Bigg(\varphi_{11}^{-1}\Big(1-\big|\varphi_{12}\big(\omega_1(X)\big)-\varphi_{12}\big(\omega_1(Y)\big)\big|\Big),\ldots,\varphi_{n1}^{-1}\Big(1-\big|\varphi_{n2}\big(\omega_n(X)\big)-\varphi_{n2}\big(\omega_n(Y)\big)\big|\Big)\Bigg)\ , \tag{5}$$

para todo $X,Y\in L^n$, es una R_{L^n} en L^n.

Nótese que es posible reescribir la anterior expresión en términos de Eq. 2 dado el caso en el que ω_i son MAPs,

$$R_{L^n}(X,Y)=\mathbf{R}\left((WX^T)^T,(WY^T)^T\right)\ , \tag{6}$$

donde W es la matriz construida con los pesos de las MAP y $\mathbf{R}$ es una secuencia de REFs en L.

Tomemos los colores crisp presentados en la Subsección II-B y compongamos las imágenes sintéticas A y B (Figuras 2a y 2b respectivamente). Ambas son la misma imagen pero una rotada con respecto a la otra, y además, la segunda contiene tres caracteres adicionales: el caracter blanco "L" a la izquierda, el caracter cian "R" a la derecha y un tercer carácter negro "C" en el centro.

Ejemplo III.4. *Las imágenes sintéticas Fig. 2a y Fig. 2b nos permiten comparar muchos de los colores crisp entre sí al usar el Alg. 1. En este ejemplo, el resultado se calcula dos veces; uno para $W=\mathbb{I}$ y otro para $W=\Gamma$, siendo*

$$\Gamma=\begin{pmatrix}0.6&0.2&0.2\\0.2&0.6&0.2\\0.2&0.2&0.6\end{pmatrix},$$

donde, en ambos casos la secuencia de REFs elegida es $\mathbf{R}=(1-|x-y|,1-|x-y|,1-|x-y|)$.

Los mapas de características resultantes se muestran en la Fig. 3. Se observa cómo al comparar C_R y C_B devuelve el mismo tono verde que al comparar C_C y C_Y para diferentes valores de W sin rotación. Esta simetría se debe a que el canal que permanece inalterado es el segundo. Véase como, tomando $C_C=(0,1,1)$ y $C_Y=(1,1,0)$ el canal con valor equivalente sigue siendo el segundo. Entonces, esto se puede interpretar como; los tonos de color del mapa resultante identifican los canales que son más equivalentes entre los pixeles comparados, haciendo coincidir el tono del pixel resultante con sus canales correspondientes cuando no hay rotación inducida por W.

Ejemplo de (RL1) es que el fondo de ambos mapas es C_W porque se compara el mismo color; en ambos casos, el color C_K está como fondo. Además, el área circundante del carácter "C" en los mapas es C_W porque en ambas imágenes de entrada se compara el color C_W. Por el contrario, teniendo en cuenta (RL2) C_K se obtiene en el mapa resultante para el carácter "C" y la parte superior del carácter "L" al comparar $\mathbf{0}$ con $\mathbf{1}$.

El color en la parte izquierda de "R" es el resultado de comparar dos pares de colores crisp complementarios (C_R-C_C) y depende de W. Para $W=\mathbb{I}$ (en la Fig. 3a), se obtiene C_K y lo mismo sucede si se comparan cualquiera de los colores crisp complementarios en el cubo RGB; C_K-C_W, C_R-C_C, C_G-C_M o C_B-C_Y. Esto implica que todos los pares crisp complementarios tienen el mismo comportamiento con respecto a la REF que va en contra de la filosofía establecida. Entonces, debido a esta razón, W no puede ser la matriz de identidad para R_{L^n}. En otras palabras, esto significa que C_K se puede obtener en más casos que comparando exclusivamente C_K con C_W, entonces, el axioma RL2 no se cumple para $W=\mathbb{I}$ (o, en cualquier caso, cuando $det(W)=1$).



(a) (b) (c)

Fig. 4: La imagen 3063 original (Fig. 4a) y su versión modificada (Fig. 4b) para usar como input del Alg. 1 en el Ejemplo III.5. El mapa resultante se muestra en la Fig. 4c.

Para concluir este trabajo, mostramos un ejemplo donde aplicamos el Alg. 1 usando R_{L^n}. Este Ejemplo III.5 es una posible aplicación del Algoritmo 1 donde se detecta un objeto en función de la diferencia de color entre las imágenes de entrada. Con este propósito, hemos tomado del Berkeley Segmentation Dataset [23] la imagen indexada como 3063 (Fig. 4a) y la hemos modificado para agregar un segundo avión en la esquina superior derecha como se muestra en la Fig. 4b.

Ejemplo III.5. *La R_{L^n} tal y como aparece en la Eq. 6 es usada para comparar mediante el Alg. 1 las imágenes de la Fig. 4 tomando como secuencia de REFs* $\mathbf{R} = (1-|x-y|, 1-|x-y|, 1-|x-y|)$. *Por simplicidad W es la misma que en el Ejemplo II.5. El mapa resultante se muestra en la Fig. 4c.*

IV. Líneas futuras

Una evidente línea futura es generar una nueva clase de operadores L^n-REF que se basen en la filosofía en la cual todos los posibles complementarios crisp sean tratados de la misma manera, por ejemplo, deberán tomar como igualmente antagónicos todos los elementos multivaluados que sean complementarios crisp y no acentuar la diferencia para el único par $\mathbf{0} - \mathbf{1}$.

En concreto, la creación de este tipo de operador nos llevaría a tener que analizar qué diferencias se podrían apreciar entre la aplicación de los operadores construidos en base a las diferentes filosofías en el procesamiento de imagen a color.

Otra posible línea futura es extender las aplicaciones a otro tipo de formato de información como pueden ser las imágenes multi e hiperespectrales.

Agradecimientos

Este trabajo ha sido respaldado por los proyectos PID2019-108392GB-I00 (AEI/10.13039/ 501100011033) de la Agencia Estatal de Investigación.

References

[1] F. Van Breugel, J. Worrell, A behavioural pseudometric for probabilistic transition systems, Theoretical Computer Science 331 (1) (2005) 115–142.
[2] C. Lopez-Molina, S. Iglesias-Rey, H. Bustince, B. De Baets, On the role of distance transformations in baddeley's delta metric, Information Sciences (2021).
[3] A. Tversky, I. Gati, Similarity, separability, and the triangle inequality., Psychological review 89 (2) (1982) 123.
[4] A. Tversky, D. H. Krantz, The dimensional representation and the metric structure of similarity data, Journal of mathematical psychology 7 (3) (1970) 572–596.
[5] I. Kramosil, J. Michálek, Fuzzy metrics and statistical metric spaces, Kybernetika 11 (5) (1975) 336–344.
[6] O. Kaleva, S. Seikkala, On fuzzy metric spaces, Fuzzy sets and systems 12 (3) (1984) 215–229.
[7] Z. Deng, Fuzzy pseudo-metric spaces, Journal of Mathematical Analysis and Applications 86 (1) (1982) 74–95.
[8] Y. Yue, F.-G. Shi, On fuzzy pseudo-metric spaces, Fuzzy Sets and Systems 161 (8) (2010) 1105–1116.
[9] H. Bustince, E. Barrenechea, M. Pagola, Restricted equivalence functions, Fuzzy Sets and Systems 157 (17) (2006) 2333–2346.
[10] J. C. Fodor, M. Roubens, Fuzzy preference modelling and multicriteria decision support, Vol. 14, Springer Science & Business Media, 1994.
[11] A. Jurio, M. Pagola, D. Paternain, C. Lopez-Molina, P. Melo-Pinto, Interval-valued restricted equivalence functions applied on clustering techniques, in: Proc. of the IFSA-EUSFLAT, 2009, pp. 831–836.
[12] C. Marco-Detchart, J. Cerron, L. De Miguel, C. Lopez-Molina, H. Bustince, M. Galar, A framework for radial data comparison and its application to fingerprint analysis, Applied Soft Computing 46 (2016) 246–259.
[13] V. Barnett, The ordering of multivariate data, Journal of the Royal Statistical Society: Series A 139 (1976) 318–355.
[14] H. Bustince, J. Montero, E. Barrenechea, M. Pagola, Semiautoduality in a restricted family of aggregation operators, Fuzzy Sets and Systems 158 (12) (2007) 1360–1377.
[15] D. Dubois, H. Prade, A review of fuzzy set aggregation connectives, Information sciences 36 (1-2) (1985) 85–121.
[16] G. J. Klir, Fuzzy sets, Uncentainty and Information (1988).
[17] H. Bustince, M. Pagola, E. Barrenechea, Construction of fuzzy indices from fuzzy di-subsethood measures: application to the global comparison of images, Information Sciences 177 (3) (2007) 906–929.
[18] H. J. Trussell, E. Saber, M. Vrhel, Color image processing: Basics and special issue overview, IEEE signal processing magazine 22 (1) (2005).
[19] T. Mélange, Fuzzy techniques for noise removal in image sequences and interval-valued fuzzy mathematical morphology, Ph.D. thesis, Ghent University (2010).
[20] G. Sharma, H. J. Trussell, Digital color imaging, IEEE transactions on image processing 6 (7) (1997) 901–932.
[21] G. Wyszecki, W. S. Stiles, Color science: concepts and methods, quantitative data and formulas (1982).
[22] J. Cohen, Visual color and color mixture: The fundamental color space, University of Illinois Press, 2001.
[23] P. Arbelaez, M. Maire, C. Fowlkes, J. Malik, Contour detection and hierarchical image segmentation, IEEE Trans. on Pattern Analysis and Machine Intelligence 33 (2011) 898–916.

XX Congreso Español sobre Tecnologías y Lógica Fuzzy

Trabajos Destacados



Computing with Comparative Linguistic Expressions and Symbolic Translation for Decision Making: ELICIT Information

1st Álvaro Labella
Department of Computer Science
University of Jaén
Jaén, Spain
alabella@ujaen.es

2nd Rosa M. Rodríguez
Department of Computer Science
University of Jaén
Jaén, Spain
rmrodrig@ujaen.es

3rd Luis Martínez
Department of Computer Science
University of Jaén
Jaén, Spain
martin@ujaen.es

***Abstract*—This is a summary of our article published in IEEE Transaction on Fuzzy Systems [1]. This article introduces a new fuzzy linguistic representation model for comparative linguistic expressions that takes advantage of the goodness of the 2-tuple linguistic representation model and improves the readability and accuracy of the results in computing with words processes, resulting the so-called extended comparative linguistic expressions with symbolic translation.**

***Index Terms*—Decision making, computing with words, symbolic translation, comparative linguistic expressions**

I. Introduction

Nowadays, *Decision Making* (DM) problems are defined in changing contexts in which uncertainty and vagueness are quite common. The *fuzzy linguistic approach* [2] has been used successfully to model such uncertainty by means of linguistic information, giving rise to the so-called *Linguistic Decision Making*. Consequently, the use of linguistic information implies to perform computations with it in order to solve decision problems under uncertainty. *Computing with Words* (CW) [3] methodology carries out processes *"where words and not numbers are used for computing"* and mimics the human beings' reasoning process in which, from linguistic premises, provides linguistic results. In this way, CW methodology guarantees interpretable results.

There are several proposal that try to follow a CW approach. One of the most widely used in DM is the 2-tuple linguistic model [4] which presents interpretable and precise linguistic results, the latter thanks to use of the symbolic translation concept. However, such results are represented by single linguistic terms that are inadequate to represent experts' hesitancy. To overcome the latter limitation, Rodríguez et al. [5] introduced the *Hesitant Fuzzy Linguistic Term Set* (HFLTS) that, together with the use of context-free grammars, allow to generate *Comparative Linguistic Expressions* (CLEs) close to the natural language used by human beings. However, the existing computational model that makes use of CLEs does not provide interpretable results.

Therefore, in order to overcome previous drawbacks, we propose a new fuzzy linguistic representation model that extends the CLEs by using the concept of symbolic translation introduced by the 2-tuple linguistic model resulting the so-called *Extended Comparative LInguistiC Expressions with SymbolIc Translation* (ELICIT) information. These expressions extend the representation of CLEs generated by a context-free grammar into a continuous domain to perform CW processes without any kind of approximation. The proposed context-free grammar to generate ELICIT information is described below:

Definition 1: [1] Let G_H be a context-free grammar and $S = \{s_0, \ldots, s_g\}$ a linguistic term set. The elements of $G_H = (V_N, V_T, I, P)$ are defined as follows.

$V_N = \{(continuous\ primary\ term),\ (composite\ term),$
$(unary\ relation), (binary\ relation), (conjunction)\}$
$V_T = \{at\ least,\ at\ most,\ between,\ and,\ (s_0, \alpha)^\gamma,$
$(s_1, \alpha)^\gamma, \ldots, (s_g, \alpha)^\gamma\}$
$I \in V_N$
$P = \{I\ ::=\ (continuous\ primary\ term)|$
$(composite\ term)$
$(composite\ term)\ ::=\ (unary\ relation)$
$(continuous\ primary\ term)|$
$(binary\ relation)(continuous\ primary\ term)$
$(conjunction)(continuous\ primary\ term)$
$(continuous\ primary\ term)\ ::=\ (s_0, \alpha)^\gamma|$
$(s_1, \alpha)^\gamma| \ldots |(s_g, \alpha)^\gamma$
$(unary\ relation)\ ::=\ at\ least|at\ most$
$(binary\ relation)\ ::=\ between$
$(conjunction)\ ::=\ and\}$

Thus, the possible ELICIT expressions generated according to the new definition of the context-free grammar are: "*at least* $(s_i, \alpha)^\gamma$", "*at most* $(s_i, \alpha)^\gamma$" and "*between* $(s_i, \alpha_1)^{\gamma_1}$ *and* $(s_j, \alpha_2)^{\gamma_2}$"

A CW approach for ELICIT information has been also proposed in [1]. Such a CW approach obtains linguistic results

modeled by ELICIT information from linguistic inputs represented by CLEs and ELICIT information. To carry out CW processes, linguistic inputs are transformed into *trapezoidal fuzzy numbers* (TrFNs), which are manipulated by means of fuzzy parametric operations. Whereas CLEs are transformed into TrFNs by means of the fuzzy envelope, such a transformation for ELICIT information is carried out by means of the function ζ^{-1}.

Definition 2: [1] Let x_{el} be an ELICIT expression and $T(a,b,c,d)$ a trapezoidal fuzzy number. The function ζ^{-1} is defined as:

$$\zeta^{-1} : x_{el} \rightarrow T(a,b,c,d) \tag{1}$$

The manipulation of the fuzzy envelopes through fuzzy parametric operations provides new fuzzy numbers noted as β. Now, it is necessary a retranslation process to obtain ELICIT information from TrFNs. This process consists of different steps:

1) *Identify the relation*: the relation is determined by the fuzzy number $\tilde{\beta}$ and the ζ function, defined as follows: *Definition 3:* Let $S = \{s_0, \ldots, s_g\}$ be a set of linguistic terms and $\tilde{\beta}$ a fuzzy number. The function ζ is given by

$$\zeta(\tilde{\beta}) = x_{el},\ where \begin{cases} x_{el} = at\ least\ (s_i,\alpha)^{\gamma}\ if\ \tilde{\beta} = T(a,b,1,1) \\ x_{el} = at\ most\ (s_i,\alpha)^{\gamma}\ if\ \tilde{\beta} = T(0,0,c,d) \\ x_{el} = between\ (s_i,\alpha_1)^{\gamma_1} and\ (s_j,\alpha_2)^{\gamma_2} \\ if\ \tilde{\beta} = T(a,b,c,d) \end{cases}$$

Henceforth, for sake of space, it is assumed that the ELICIT expression is "$\boldsymbol{between\ (s_i,\alpha_1)^{\gamma_1}\ and\ (s_j,\alpha_2)^{\gamma_2}}$".

2) *2-tuple linguistic terms computation*: the process of obtaining the two continuous primary terms $(s_i,\alpha_1)^{\gamma_1}$ and $(s_j,\alpha_2)^{\gamma_2}$ is divided into different steps:
   a) *Compute linguistic terms*: select the linguistic terms s_i and $s_j \in S,\ i,\ j \in \{0,\ldots g\}$, whose distance between the coordinates x of their respective centroids, $\overline{x}_i$ and $\overline{x}_j$, and the points b and c belonging to $\tilde{\beta}$ is minimal.

$$\begin{aligned} i &= \arg\min_h |b - \overline{x}_h|,\ h \in \{0,\ldots,g\} \\ j &= \arg\min_h |c - \overline{x}_h|,\ h \in \{0,\ldots,g\} \end{aligned} \tag{2}$$

   When this step finishes, the ELICIT expression so far is "$\boldsymbol{between\ (s_i,?)^{?}\ and\ (s_j,?)^{?}}$".
   b) *Compute symbolic translations*: according to [4], $1/2g$ represents the distance equivalent to a symbolic translation equal to 0.5 in S, where $g+1$ is the cardinality of S:

$$\begin{aligned} \alpha_1 &= g \cdot (b - \overline{x}_i) \quad \alpha_1 \in [-0.5, 0.5) \\ \alpha_2 &= g \cdot (c - \overline{x}_j) \quad \alpha_2 \in [-0.5, 0.5) \end{aligned} \tag{3}$$

   When this step finishes, the ELICIT expression so far is "$\boldsymbol{between\ (s_i,\alpha_1)^{?}\ and\ (s_j,\alpha_2)^{?}}$".
3) *Compute adjustments*: the *adjustment* is an additional parameter included in the ELICIT expression, which allows to keep information related to the fuzzy number $\tilde{\beta}$. This parameter will be used to obtain the fuzzy number $\tilde{\beta}$ from an ELICIT expression by using its inverse function, ζ^{-1}. The steps to compute the adjustments for the ELICIT expression are:
   a) *Compute HFLTS*: the HFLTS of an ELICIT expression whose relation is *between* would be composed by:

$$E_{ELICIT}(between\ (s_i,\alpha)\ and\ (s_j,\alpha)) = \{s_k \mid (s_i,\alpha)\ and\ (s_j,\alpha), s_i < s_k < s_j\ where\ s_k \in S\}$$

   b) *Compute fuzzy envelope*: the fuzzy envelope, $T_{ELICIT} = T(a',b',c',d')$, of the former HFLTS is computed.
   c) *Compute adjustments* γ_1 *and* γ_2: the adjustments γ_1 and γ_2 are determined by the subtraction between the points a and d of $\tilde{\beta} = T(a,b,c,d)$ and the points a' and d' of $T(a',b',c',d')$, so that:

$$\begin{aligned} \gamma_1 &= a - a' \quad \gamma_1 \in [-1,1] \\ \gamma_2 &= d - d' \quad \gamma_2 \in [-1,1] \end{aligned} \tag{4}$$

   When this step finishes, the ELICIT expression is completed "$\boldsymbol{between\ (s_i,\alpha_1)^{\gamma_1}\ and\ (s_j,\alpha_2)^{\gamma_2}}$".

II. Conclusions

The need of a new fuzzy linguistic representation model that overcomes the existing limitations in previous linguistic models either from the point of view of interpretability or/and accuracy has resulted in the ELICIT representation model and its CW approach. This new linguistic model makes use of the ELICIT information, CLEs extended to a continuous domain by means of the symbolic translation concept. In this way, it is possible to carry out CW processes with high accuracy and interpretability.

As future works, we will study the definition of new aggregation operators for ELICIT information. Another aim is to apply this new type of information to consensus reaching processes.

Acknowledgment

This work is partially supported by the Spanish Ministry of Economy and Competitiveness through the Spanish National Research Project PGC2018-099402-B-I00 and the Postdoctoral fellow Ramón y Cajal (RYC-2017-21978).

References

[1] Á. Labella, R. M. Rodríguez, and L. Martínez, "Computing with comparative linguistic expressions and symbolic translation for decision making: Elicit information," *IEEE Transactions on Fuzzy Systems*, vol. 28, no. 10, pp. 2510–2522, 2019.

[2] L. A. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning (i)," *Information Sciences*, vol. 8, no. 3, pp. 199 – 249, 1975.

[3] ——, "Fuzzy logic = computing with words," *IEEE Transactions on Fuzzy Systems*, vol. 4, no. 2, pp. 103–111, 1996.

[4] L. Martínez, R. M. Rodríguez, and F. Herrera, *The 2-tuple Linguistic Model*. Springer International Publishing, 2015.

[5] R. M. Rodríguez, L. Martínez, and F. Herrera, "A group decision making model dealing with comparative linguistic expressions based on hesitant fuzzy linguistic term sets," *Information Sciences*, vol. 241, no. 1, pp. 28–42, 2013.

Aggregation of fuzzy quasi-metrics

Tatiana Pedraza
Instituto Universitario de Matemática Pura y Aplicada
Universitat Politècnica de València
Valencia, Spain
tapedraz@mat.upv.es

Jesús Rodríguez-López
Instituto Universitario de Matemática Pura y Aplicada
Universitat Politècnica de València
Valencia, Spain
jrlopez@mat.upv.es

Óscar Valero
Departament de Ciències Matemàtiques i Informàtica
Universitat de les Illes Balears
Palma, Spain
o.valero@uib.es

***Abstract*—In this work we will present some results about the aggregation of fuzzy (quasi-)metrics that appear in [9]. Concretely, we provide a characterization of functions that merge a collection of fuzzy (quasi-)metrics into a single one in terms of $*$-triangular triplets, isotonicity and $*$-supmultiplicativity, where $*$ is a t-norm. We also show that, in contrast to the crisp case, this characterization does not depend on the symmetry of the fuzzy quasi-metrics. We also stress that these results are not only interesting from the aggregation theory viewpoint but also because they allow to generate examples of fuzzy (quasi-)metric spaces that are not easy to obtain. Moreover, from our results we can infer others about the aggregation of fuzzy preorders and indistinguishability operators.**

I. Introduction

The problem of aggregating several structures of the same type into a single one has received a lot of attention in the last years. In this way, we can find results about the aggregation of: metrics [1]; quasi-metrics [7]; norms [4]; asymmetric norms [5]; fuzzy binary relations [11], [2]; indistinguishability operators [6]; etc. In the following, we will present some results about the aggregation of an important fuzzy structure: the fuzzy quasi-metrics. This concept has its origins in the probabilistic metric spaces introduced by Menger [8] in 1942 who gave a probabilistic interpretation of the concept of distances and proposed to associate a distribution function with a pair of elements, instead of associating a number.

In some sense, fuzzy binary relations can be considered as a particular class of fuzzy (quasi-)metrics. We recall its definition as well as one of its most important classes.

Definition 1 ([10])**.** *A* fuzzy binary relation *on a nonempty set X is a map $E : X \times X \to [0,1]$.*
If a fuzzy binary relation E on X satisfies for all $x, y, z \in X$:

- $E(x,x) = 1$ *(*reflexivity*)*
- $E(x,y) = E(y,x)$ *(*symmetry*)*
- $E(x,y) * E(y,z) \leq E(x,z)$ *($*$-transitivity)*

*where $*$ is a triangular norm, then it is called an $*$-*indistinguishability operator.

In the literature we can find several papers [2], [3], [6], [11] studying functions which preserve $*$-transitivity of fuzzy binary relations in the following two, a priori, different senses:

Definition 2 (cf. [6], [11])**.** *Let I be a set of indices and $F : [0,1]^I \to [0,1]$ be a function. We say that:*

- F preserves $*$-transitivity of fuzzy binary relations on products *if whenever $\{(X_i, E_i) : i \in I\}$ is a family of nonempty sets X_i endowed with $*$-transitive fuzzy binary relations E_i for all $i \in I$, then $F \circ \widetilde{\boldsymbol{E}}$ is an $*$-transitive fuzzy binary relation on $\prod_{i\in I} X_i$ where $\widetilde{\boldsymbol{E}} : (\prod_{i\in I} X_i)^2 \to [0,1]^I$ is given by*
$$\widetilde{\boldsymbol{E}}(\boldsymbol{a}, \boldsymbol{b})_i = E_i(\boldsymbol{a}_i, \boldsymbol{b}_i) \qquad \text{for all } i \in I.$$
- F preserves $*$-transitivity of fuzzy binary relations on sets *if whenever $\{E_i : i \in I\}$ is a family of $*$-transitive fuzzy binary relations E_i on a fixed nonempty set X for all $i \in I$, then $F \circ \boldsymbol{E}$ is an $*$-transitive fuzzy binary relation on X where $\boldsymbol{E} : X^2 \to [0,1]^I$ is given by*
$$\boldsymbol{E}(a,b)_i = E_i(a,b) \qquad \text{for all } i \in I.$$

For studying functions which aggregate fuzzy quasi-metrics it is important to take into account the characterization of those functions which preserve $*$- transitivity of fuzzy binary relations, since fuzzy (quasi-)metrics satisfy a property near to $*$-transitivity. Surprisingly, we can prove that there is no difference between those functions which preserve $*$-transitivity of fuzzy binary relations on products and those which preserve $*$-transitivity of fuzzy binary relations on sets.

Definition 3. *Let $*$ be a t-norm and I be a set of indices. A triplet $(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}) \in ([0,1]^I)^3$ is said to be* asymmetric $*$-triangular *if $\boldsymbol{a}_i * \boldsymbol{b}_i \leq \boldsymbol{c}_i$ for all $i \in I$.*

Definition 4. *Let $*$ be a t-norm and I be a set of indices. A function $F : [0,1]^I \to [0,1]$* preserves asymmetric $*$-triangular triplets *if $(F(\boldsymbol{a}), F(\boldsymbol{b}), F(\boldsymbol{c}))$ is an asymmetric $*$-triangular triplet whenever $(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c})$ so is, where $\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c} \in [0,1]^I$.*

Proposition 5 ([9])**.** *Let $F : [0,1]^I \to [0,1]$ be a function and $*$ be a t-norm. The following statements are equivalent:*

1) *F preserves $*$-transitivity of fuzzy binary relations on products;*
2) *F preserves $*$-transitivity of fuzzy binary relations on sets;*
3) *F preserves asymmetric $*$-triangular triplets.*

II. Aggregation of fuzzy quasi-metrics

Definition 6. *A* fuzzy quasi-metric *(in the sense of Kramosil and Michalek) on a nonempty set X is a pair $(M,*)$ such that $*$ is a t-norm and M is a fuzzy set in $X\times X\times[0,+\infty)$ such that for every $x,y,z\in X$ and $t,s>0$ it verifies:*

- $M(x,y,0)=0$;
- $M(x,y,t)=M(y,x,t)=1$ *for all* $t>0$ *if and only if* $x=y$;
- $M(x,y,t)*M(y,z,s)\leq M(x,z,t+s)$;
- $M(x,y,\cdot):[0,\infty)\to[0,1]$ *is left-continuous.*

If a fuzzy quasi-metric $(M,*)$ also satisfies

- $M(x,y,t)=M(y,x,t)$

for all $x,y\in X$ and all $t\geq 0$ then $(M,*)$ is said to be a fuzzy metric on X.
A fuzzy (quasi-)metric *space is a triple $(X,M,*)$ such that X is a nonempty set and $(M,*)$ is a fuzzy (quasi-)metric on X.*

Definition 7. *A function $F:[0,1]^I\to[0,1]$ is said to be:*

- *a* fuzzy (quasi-)metric aggregation function on products *if whenever $*$ is a t-norm and $\{(X_i,M_i,*):i\in I\}$ is a family of fuzzy (quasi-)metric spaces then $(F\circ\widetilde{\boldsymbol{M}},*)$ is a fuzzy (quasi-)metric on $\prod_{i\in I}X_i$ where $\widetilde{\boldsymbol{M}}:(\prod_{i\in I}X_i)^2\times[0,+\infty)\to[0,1]^I$ is given by*

$$(\widetilde{\boldsymbol{M}}(\boldsymbol{x},\boldsymbol{y},t))_i=M_i(x_i,y_i,t)$$

  for every $\boldsymbol{x},\boldsymbol{y}\in\prod_{i\in I}X_i$ and $t\geq 0$.
  If F only satisfies the above condition for a fixed t-norm $*$ then it is said to be an $*$-fuzzy (quasi-)metric aggregation function on products.
- *a* fuzzy (quasi-)metric aggregation function on sets *if whenever $*$ is a t-norm and $\{(M_i,*):i\in I\}$ is a family of fuzzy (quasi-)metrics on the same set X then $(F\circ\boldsymbol{M},*)$ is a fuzzy (quasi-)metric on X where $\boldsymbol{M}:X^2\times[0,+\infty)\to[0,1]^I$ is given by*

$$(\boldsymbol{M}(x,y,t))_i=M_i(x,y,t)$$

  for every $x,y\in X$ and $t\geq 0$.
  If F only satisfies the above condition for a fixed t-norm $*$ then it is said to be an $*$-fuzzy (quasi-)metric aggregation function on sets.

The next result characterizes those functions which merge an arbitrary family of fuzzy quasi-metrics into a single one. Surprisingly, these functions are exactly those which merge a family of fuzzy metrics into a fuzzy metric, which is not true in the crisp case [7].

Definition 8 (cf. [11])**.** *A function $F:[0,1]^I\to[0,1]$ is said to be* $*$-supmultiplicative *for a t-norm $*$ if for all $\boldsymbol{x},\boldsymbol{y}\in[0,1]^I$ then*

$$F(\boldsymbol{x})*F(\boldsymbol{y})\leq F(\boldsymbol{x}*^I\boldsymbol{y})$$

where $\boldsymbol{x}*^I\boldsymbol{y}\in[0,1]^I$ is given by $(\boldsymbol{x}*^I\boldsymbol{y})_i=\boldsymbol{x}_i*\boldsymbol{y}_i$ for all $i\in I$.

Theorem 9 ([9])**.** *Let $F:[0,1]^I\to[0,1]$ be a function and $*$ be a t-norm. The following statements are equivalent:*

1) *F is a ($*$-)fuzzy quasi-metric aggregation function on products;*
2) *F is a ($*$-)fuzzy metric aggregation function on products;*
3) *F is isotone, ($*$-)supmultiplicative, left-continuous, $F(\mathbf{0})=0$ and $F^{-1}(1)=\mathbf{1}$;*
4) *$F(\mathbf{0})=0$, F is left-continuous, $F^{-1}(1)=\mathbf{1}$ and F preserves asymmetric ($*$-)triangular triplets.*

Theorem 10 ([9])**.** *Let $F:[0,1]^I\to[0,1]$ be a function and $*$ be a t-norm. The following statements are equivalent:*

1) *F is a ($*$-)fuzzy quasi-metric aggregation function on sets;*
2) *F is a ($*$-)fuzzy metric aggregation function on sets;*
3) *F is isotone, ($*$-)supmultiplicative, left-continuous, $F(\mathbf{0})=0$ and if $\{\boldsymbol{x}_n:n\in\mathbb{N}\}\subseteq F^{-1}(1)$ there exists $i\in I$ such that $(\boldsymbol{x}_n)_i=1$ for all $n\in\mathbb{N}$;*
4) *$F(\mathbf{0})=0$, $F(\mathbf{1})=1$, F is left-continuous, F preserves asymmetric ($*$-)triangular triplets and if $\{\boldsymbol{x}_n:n\in\mathbb{N}\}\subseteq F^{-1}(1)$ there exists $i\in I$ such that $(\boldsymbol{x}_n)_i=1$ for all $n\in\mathbb{N}$.*

More results about this topic can be consulted in [9].

Acknowledgment

The two last authors acknowledge financial support from FEDER/Ministerio de Ciencia, Innovación y Universidades-Agencia Estatal de Investigación-Proyecto PGC2018-095709-B-C21.

References

[1] J. Doboš, "Metric preserving functions," Štroffek, Košice, 1998.
[2] J. Drewniak and U. Dudziak, "Aggregation in classes of fuzzy relations," *Stud. Math.*, vol. 5, pp. 33–43, 2006.
[3] U. Dudziak, "Preservation of *t*-norm and *t*-conorm based properties of fuzzy relations during aggregation process," *Proc. of 8th Conference of the European Society for Fuzzy Logic and Technology* (EUSFLAT 2013), 2013, pp. 376–383.
[4] I. Herburt and M. Moszyńska, "On metric products," *Colloq. Math.*, vol. 62, pp. 121–133, 1991.
[5] J. Martín, G. Mayor, and O. Valero, "On aggregation of normed structures," *Math. Comput. Modelling*, vol. 54, pp. 815–827, 2011.
[6] G. Mayor and J. Recasens, "Preserving T-transitivity," in *Artificial Intelligence Research and Development*, 2016, vol. 288, pp. 79–87.
[7] G. Mayor and O. Valero, "Aggregation of asymmetric distances in Computer Science," *Information Sci.*, vol. 180, pp. 803–812, 2010.
[8] K. Menger, "Statistical metrices," *Proc. Natl. Acad. Sci. USA*, vol. 28, pp. 535–537, 1942.
[9] T. Pedraza, J. Rodríguez-López and O. Valero, "Aggregation of fuzzy quasi-metrics," *Information Sci.*, to appear.
[10] J. Recasens, *Indistinguishability operators. Modelling fuzzy equalities and fuzzy equivalence relations*. Springer-Verlag, 2010.
[11] S. Saminger, R. Mesiar, and U. Bodenhofer, "Domination of aggregation operators and preservation of transitivity," *Internat. J. Uncertain. Fuzziness Knowledge-Based Systems*, vol. 10, pp. 11–35, 2002.

A graded notion of quasi-intents

Manuel Ojeda-Hernndez
Dept. Applied Mathematics
University of Mlaga
Mlaga, Spain
manuojeda@uma.es

Inma P. Cabrera
Dept. Applied Mathematics
University of Mlaga
Mlaga, Spain
ipcabrera@uma.es

Pablo Cordero
Dept. Applied Mathematics
University of Mlaga
Mlaga, Spain
pcordero@uma.es

***Abstract*—The notion of quasi-closed element is extended to fuzzy posets in two stages: First, in the crisp style, in which each element in a given universe either is quasi-closed or it is not. Second, in the graded style by defining a degree to which an element is quasi-closed. We discuss the different possible definitions and compare them with each other. Finally, we show that the most general one has good properties that can be used when we have a complete fuzzy lattice as a frame.**

***Index Terms*—Quasi-closed element, fuzzy poset, closure operator.**

I. Introduction

In this work, the research in [1] is revisited. The goal is to discuss which is the appropriate generalization of the notion of quasi-closedness when working on fuzzy posets. In the classical case, this notion is key to knowledge representation ensuring non-redundancy. However, obtaining an adequate generalization for fuzzy environments that guarantees similar properties, remains an open problem.

A wide variety of generalizations to the fuzzy framework of the notion of implication (and logics for reasoning about them) can be found in the literature, see for example [2]. In [3], the authors include a general framework for these generalizations. All the results on pseudo-closed elements for the fuzzy case have been obtained by using a recursive definition of pseudo-closedness. In the classical case, there exists an equivalent definition based on the notion of quasi-closed element. In this paper, we aim to generalize such notion to the fuzzy framework, which, in the short term, may provide with an alternative definition of pseudo-closed element, as a starting point for a new approach in the study of bases in a fuzzy environment.

As stated in [4], "clearly, in the graded setting, the topics related to non-redundancy and minimality of bases are considerably more involved than in the classic setting and further investigation focused on theory, algorithms, and experiments is needed".

II. Preliminaries

Throughout this paper, let $\mathbb{L} = (L, \wedge, \vee, \otimes, \to, 0, 1)$ be a complete residuated lattice. A non-empty set A with a binary $\mathbb{L}$-relation ρ on A, is said to be a *fuzzy poset* if ρ is a *fuzzy order*, i.e. if ρ is reflexive, antisymmetric and transitive.

The notions of lower (resp. upper) bound and infimum (resp. supremum) used in this work are the ones presented in [5].

Theorem 1: Let $\mathbb{A} = (A, \rho)$ be a fuzzy poset and $X \in L^A$. An element $a \in A$ is supremum (resp. infimum) of X if and only if

$$\rho(a, x) = X^{\rho}(x) \qquad (\text{resp. } \rho(x, a) = X_{\rho}(x)).$$

It is not difficult to see that, if a supremum (resp. infimum) of X exists, it is unique. We will denote it by $\bigsqcup X$ (resp. $\bigsqcap X$).

Definition 2 ([6]): We say that a fuzzy poset (A, ρ) is a complete fuzzy lattice if every fuzzy subset $X \in L^A$ has supremum and infimum.

We conclude this section with the usual definition of closure operator on a fuzzy poset.

Definition 3: Given a fuzzy poset $\mathbb{A} = (A, \rho)$, a mapping $\mathtt{c}\colon A \to A$ is said to be a *closure operator* on $\mathbb{A}$ if the following conditions hold:

1) $\rho(a, b) \leq \rho(\mathtt{c}(a), \mathtt{c}(b))$, for all $a, b \in A$ (isotony)
2) $\rho(a, \mathtt{c}(a)) = 1$, for all $a \in A$ (inflationarity)
3) $\rho(\mathtt{c}(\mathtt{c}(a)), \mathtt{c}(a)) = 1$, for all $a \in A$ (idempotency)

Definition 4: Let $\mathtt{c}\colon A \to A$ be a closure operator on a fuzzy poset (A, ρ) and X be an $\mathbb{L}$-subset of A. The closure of X wrt $\mathtt{c}$ is the $\mathbb{L}$-set defined by

$$\mathtt{c}(X)(a) = \bigvee_{x \in \mathtt{c}^{-1}(a)} X(x), \qquad \text{for all } a \in A.$$

III. Generalizing the notion of quasi-closed element to fuzzy posets

The aim of this section is to analyse possible generalisations to fuzzy posets of the classical notion of quasi-closed element. We begin by recalling the definition in the case of crisp posets [7]. Throughout this section $\mathbb{A} = (A, \rho)$ is a fuzzy poset and $\mathtt{c}$ is a closure operator on $\mathbb{A}$.

There are four equivalent properties that define quasi-closedness in the crisp case. The direct extensions of each one these statements to a fuzzy setting are the following.

(I) $\rho(a, q) \leq \rho(\mathtt{c}(a), q) \vee (\mathtt{c}(a) \approx \mathtt{c}(q))$, for all $a \in A$.
(II) $\rho(a, q) \leq \rho(\mathtt{c}(a), q) \vee \rho(\mathtt{c}(q), \mathtt{c}(a))$, for all $a \in A$.
(III) $\rho(a, q) \otimes \neg\rho(\mathtt{c}(q), \mathtt{c}(a)) \leq \rho(\mathtt{c}(a), q)$, for all $a \in A$.
(IV) $\rho(a, q) \otimes \neg\rho(q, a) \otimes \rho(\mathtt{c}(a), \mathtt{c}(q)) \otimes \neg\rho(\mathtt{c}(q), \mathtt{c}(a)) \leq \rho(\mathtt{c}(a), q)$, for all $a \in A$.

These relation among these statements are in the following proposition.

Proposition 5:

- **(I)** implies **(II)**.
- **(II)** implies **(III)**.
- **(III)** implies **(IV)**.

None of the converses hold.

Proposition 6: Given a closure operator c on a (crisp) poset, any quasi-closed element with respect to c satisfies condition **(IV)**.

We adopt the most general of the statements as the definition of quasi-closed element wrt a closure operator.

Definition 7: Given a closure operator c on a fuzzy poset (A, ρ), an element $q \in A$ is said to be *quasi-closed* (with respect to c) if it satisfies statement **(IV)**.

Theorem 8:

1) Every closed element is quasi-closed.
2) There exist quasi-closed elements which are not closed.

IV. Quasi-closed elements in graded setting

In the previous section, an element either was quasi-closed or not. In this section, we define the degree to which an element is quasi-closed. First, to ease the reading of the definitions and properties, we introduce the following notation.

Notation 1: Given a closure operator c on a fuzzy poset (A, ρ) and $q \in A$, we use X_q to denote the $\mathbb{L}$-set with membership function defined as follows:

$$X_q(a) = \rho(a, q) \otimes \neg\rho(q, a) \otimes \rho(\mathsf{c}(a), \mathsf{c}(q)) \otimes \neg\rho(\mathsf{c}(q), \mathsf{c}(a)).$$

With this notation, an element q is quasi-closed iff $X_q(a) \leq \rho(\mathsf{c}(a), q)$, for all $a \in A$.

Definition 9: Given a closure operator c on a fuzzy poset (A, ρ), for any $q \in A$, we define the *degree in which q is quasi-closed* as follows

$$QC(q) = \bigwedge_{x \in A} \left[X_q(x) \to \rho(\mathsf{c}(x), q)\right].$$

Theorem 10: Let c be a closure operator on a fuzzy poset (A, ρ) and $q \in A$. Then, $QC(q) = 1$ if and only if q is quasi-closed.

In the classical setting, there is an if-and-only-if condition for a set to be quasi-closed based on an operator that is usually denoted by $^\circ$. We can do an analogous characterization here.

Definition 11: Let c be a closure operator on a complete fuzzy lattice (A, ρ) and $q \in A$. We define the element q° as follows:

$$q^\circ = q \sqcup \bigsqcup \mathsf{c}(X_q).$$

This is not a quasi-closed element in general, not even in the crisp case.

Theorem 12: Let c be a closure operator on a complete fuzzy lattice (A, ρ). Then, $QC(q) = \rho(q^\circ, q)$, for all $q \in A$.

Corollary 13: Let c be a closure operator on a complete fuzzy lattice (A, ρ). An element $q \in A$ is quasi-closed wrt c if and only if $q^\circ = q$.

V. Conclusions and further work

We have presented a fuzzy definition of quasi-closed elements in the frame of fuzzy posets and checked its properties. On the other hand, we have extended the definition to graded setting and we have proved that it extends the classical results that are necessary for its effective use in the search for bases of implications or if-then rules in the fuzzy frame.

In a next step, we will study aspects related to the computability of quasi-closed elements looking for necessary and sufficient conditions to ensure that they can be calculated efficiently. As further work, we will generalize the notion of pseudo-closed element and compare the definition obtained with the recursive one proposed in [4]. In addition, we will study whether the bases of implications with pseudo-closed premises have the desired properties of completeness, non-redundancy and minimality. We will also consider whether this work can be extended to the multi-adjoint concept lattice framework [8].

Acknowledgment

This work has been partially supported by the FPU19/01467 internship (Spanish MCIU) and the Spanish research projects with reference TIN2017-89023-P (MCIU/AEI/FEDER), PGC2018-095869-B-I00 and UMA2018FEDERJA001 (JA/UMA/FEDER).

References

[1] M. Ojeda-Hernández, I. P. Cabrera, and P. Cordero, "Quasi-closed elements in fuzzy posets," *Journal of Computational and Applied Mathematics*, vol. In Press, pp. 113–390, 2021.

[2] L. Jezková, P. Cordero, and M. Enciso, "Fuzzy functional dependencies: A comparative survey," *Fuzzy Sets Syst.*, vol. 317, pp. 88–120, 2017.

[3] P. Cordero, M. Enciso, A. Mora, and V. Vychodil, "Parameterized simplification logic: Reasoning with implications and classes of closure operators," *Int. J. General Systems*, vol. In press, pp. 1–18, 2020.

[4] V. Vychodil, "Computing sets of graded attribute implications with witnessed non-redundancy," *Information Sciences*, vol. 351, pp. 90 – 100, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0020025516301451

[5] J. Konecny and M. Krupka, "Complete relations on fuzzy complete lattices," *Fuzzy Sets and Systems*, vol. 320, pp. 64 – 80, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0165011416302718

[6] R. Bělohlvek, "Concept lattices and order in fuzzy logic," *Annals of Pure and Applied Logic*, vol. 128, no. 1, pp. 277 – 298, 2004.

[7] A. Day, "The lattice theory of functional dependencies and normal decompositions," *Int. J. Algebra and Computation*, vol. 2, pp. 409–431, 1992.

[8] M. E. Cornejo, J. Medina, and E. Ramírez-Poussa, "Attribute and size reduction mechanisms in multi-adjoint concept lattices," *Journal of Computational and Applied Mathematics*, vol. 318, pp. 388 – 402, 2017.

Community Detection Problem Based on Polarization Measures: An Application to Twitter. The COVID-19 Case in Spain. A Review

Inmaculada Gutiérrez*†, Juan Antonio Guevara *†, Daniel Gómez *‡, Javier Castro*¶ and Rosa Espínola*‖
*Faculty of Statistics
Complutense University Puerta de Hierro, s/n, 28040 Madrid, Spain
† Email: inmaguti@ucm.es and juanguev@ucm.es
§Instituto de Evaluación Sanitaria, Complutense University, 28040 Madrid, Spain

***Abstract*—In this paper, we address one of the most important topics in the field of Social Networks Analysis: the community detection problem with additional information. That additional information is modeled by a fuzzy measure that represents the risk of polarization. Adding this type of information to the community detection problem makes it more realistic, as a community is more likely to be defined if the corresponding elements are willing to maintain a peaceful dialogue. Hereafter, we work in a real case obtained from Twitter, concerning the political position against the Spanish government. We analyze how the partitions obtained change when some additional information related to how polarized that society is, is added to the problem.**

***Index Terms*—networks; community detection; extended fuzzy graphs; polarization; fuzzy sets; ordinal variation**

I. Introduction

In this document, we review our paper [1] in which we addressed one of the hottest problems in the Social Networks Analysis (SNA) field: the community detection problem. We go beyond the "classic" approach of this problem, based on the crisp connections between the elements which are defined by the edges of the graph. Our main contribution is the incorporation of some additional information modeled by a fuzzy measure.

In particular, our background is related to Polarization. In broader terms, Polarization can be understood as the split of a given society into two different and opposite groups with significant size along an attitudinal axis [2]. In this work, we recall the concept of Polarization based on fuzzy sets developed in [3], where Guevara et al. introduced a Polarization measure based on the fuzzy set approach, the JDJ_{pol}. It uses the membership degree of each individual to the poles and some aggregation operators to measure the risk of polarization of a group. We consider as an important matter to include the concept of Polarization into the field of SNA and to community detection problems due that Polarization and communication are strongly related. The way in which a community is detected in a social context should fit reality not only based on the interactions between their nodes but in its attitudinal or ideology coherence as well.

On the basis of the JDJ_{pol}, we work with (non) polarization fuzzy measures [4], which allow us to measure the ability for peaceful dialog of a society.

Then, we introduce the non polarization extended fuzzy graph (non-polarization EFG). It is characterized on the basis of the extended fuzzy graph (EFG) [5]. Given a crisp graph $G = (V, E)$ and a fuzzy measure $\mu : 2^V \to [0, 1]$ defined on the set of nodes, the triplet $\widetilde{G} = (V, E, \mu)$ is said to be an extended fuzzy graph. As it is pointed in [6], fuzzy/capacity measures are fundamental in modeling dependencies among the inputs. Then, with the combination of the ability of the graph to model connections between elements, and the ability of the fuzzy measures to handle the capacity related to any set of elements, we can represent situations in which more than two nodes are implied, independent of the way they are connected through the graph by using an EFG.

Finally, we work in a specific application of the non-polarization EFG: the community detection problem. Once we develop a methodology to find groups in a graph paying attention to how polarized the society is, we work in a real case obtained from Twitter and related to the hottest topic in the last 2020: the Covid-19 pandemic. In particular, we analyze the position of people against the Spanish government during this period of crisis.

II. Polarization fuzzy measures and Polarization extended fuzzy graph

Our first goal is the characterization of a fuzzy measure obtained from the JDJ_{pol}, μ_{P^-}. We emphasize on the fact the μ_{P^-} can be re-formulated as a summation which involves the elements of a matrix P^- which is symmetric, non-negative, 1-normalized and whose main diagonal is null. Because of the interpretation of JDJ_{pol}, P^-_{ij} represents the risk of conflict concerning the elements i and j. So that, μ_{P^-} represents the capacity of the elements to argue, to trigger conflict and arguments. Hence, it is a recommended model to properly represent the discrepancy or distance between individuals.

Because of the interpretation of the measure JDJ_{pol}, its negation, $\widetilde{JDJ_{pol}}$ can be understood as the minimum risk of polarization for a given population or community. Then, from

$\widetilde{JDJ}_{pol}$, we define the matrix P^+, non negative, symmetric, 1-normalized and with main diagonal null. It can be somehow understood as an "affinity" matrix [5], from which we can define a measure which represents the capacity of the elements of a set to peacefully dialogue without risk of Polarization, μ_{P^+}. Because of the properties of P^+, we can affirm that μ_{P^+} is a $2-$additive non-polarization fuzzy measure [7].

Then, we define a new representation model: the non polarization extended fuzzy graph. It is a crisp graph together with a non-polarization fuzzy measure, the triplet $\widetilde{G} = (V, E, \mu_{P^+})$.

III. Community Detection Problem Based on Polarization Measures

We approach the community detection problem based on fuzzy measures including this information inspired by the idea developed in the Additional Louvain algorithm (see in [7]), based on the Louvain algorithm [8]. The key point is to distinguish two different roles within the input parameters: one of them, to establish the neighbor relations, and the other, to calculate the variation of the modularity. The first role will be played by the adjacency matrix of the graph, A, so that only those nodes that are connected in G can be in the same group. On the other hand, we suggest to consider a combination of the two components of the non-polarization extended fuzzy graph $\widetilde{G}$ as basis to calculate the variation of modularity, in order to incorporate the additional information. Then, having a crisp graph, the two membership functions, and a grouping, an overlapping and a negation operators, it can be obtained a non-polarization EFG, $\widetilde{G} = (V, E, \mu_P)$, where $\mu_P = \mu_+$. In order to simplify the management of the synergies between elements, it is characterized the weighted graph associated with a fuzzy measure, G_{μ_P}, particularly, the one associated with μ_P. Being ξ an aggregation operator, $Sh_i(\mu_P)$ and $Sh_i^j(\mu_P)$ the Shapley index of i when it is in a coalition with all the elements of V and $V\{j\}$ respectively, the graph G_{μ_P} is that whose adjacency matrix is F, where

$$F_{ij} = \xi\left(Sh_i(\mu_p) - Sh_i^j(\mu_p), Sh_j(\mu_p) - Sh_j^i(\mu_p)\right) \quad (1)$$

In our specific proposal, we suggest summarizing the non-polarization fuzzy measure μ_P into the matrix F, with adjacency of its associated weighted graph defining the Polarization Louvain algorithm to detect communities. We combine the matrices A and F by means of a linear combination $(\theta(A, F) = \gamma A + (1-\gamma)F)$ using the parameter γ to assign a weight or importance to each component of the $\widetilde{G}$. Note that when $\gamma = 1$ the additional information is not considered.

IV. A Real Case: The Impact of the COVID-19 Pandemic in the Organization of the People

The nodes and theirs relations considered in this work have been obtained from the social network Twitter, particularly from some posts recorded along the state of alarm imposed by the central government in Spain. All data downloaded relate to the COVID-19 pandemic. The experiment design and all the process are detailed in [1]. A random sample of tweets were labeled by an expert in order to label them as supporters or detractors towards the Spanish Government. Then, machine learning algorithms were applied to generalize the knowledge to all tweets. Support Vector Machines showed the best performance so we used that probability of being a supporter or detractor as membership degree values to compute JDJ_{pol}. Due that we labeled tweets but not users we computed the average probability of all messages posted by a given user in order to obtain the probability to be a supporter and detractor of that user. We apply the Polarization Louvain algorithm to find communities in the non-polarization extended fuzzy graph $\widetilde{G} = (V, E, \mu_P)$. We vary the parameter γ which allow us to control the importance of that extra knowledge (Polarization values between two individuals). The partition which shows the best modularity value is considered as optimal.

To compare the Polarization Louvain algorithm with the traditional Louvain algorithm we show an example of how two pairs of nodes which should belong to the same communities, respectively, are split into four different communities with the Louvain algorithm. On one hand, we have nodes "38" and "115", both left-wing political parties that teamed back in march 2019. On the other hand, we have nodes "76" , a right-wing political party, and "203", a member of this political group. After applying the Polarization Louvain algorithm, those pairs are clustered into the same communities (see the original paper [1] for illustrations and more details).

Acknowledgment

This research has been partially supported by the Government of Spain, Grant Plan Nacional de I+D+i, MTM2015-70550-P, PGC2018096509-B-I00, TIN2015-66471-P and PID2019-106254RB-I00, and the CT17/17-CT18/17.

References

[1] Gutiérrez, I.; Guevara, J.A.; Gómez, D.; Castro, J.; Espínola, R. Community Detection Problem Based on Polarization Measures: An Application to Twitter: The COVID-19 Case in Spain. *Mathematics.* **2021**, *9*, 443.

[2] Esteban, J.M.; Ray, D. On the measurement of polarization. *Econom. J. Econom. Soc.* **1994**, *62*, 819–851.

[3] Guevara, J.A.; Gómez, D.; Robles, J.M.; Montero, J. Measuring Polarization: A Fuzzy Set Theoretical Approach. In Proceedings of the International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, Lisbon, Portugal, 15–19 June 2020; Springer: Berlin, Germany, **2020**; pp. 510–522.

[4] Sugeno, M. Fuzzy measures and fuzzy integrals: A survey. *Fuzzy Automata and Decision Processes.* **1977**, *78*, 89-102.

[5] Gutiérrez, I.; Gómez, D.; Castro, J.; Espínola, R. A New Community Detection Algorithm Based on Fuzzy Measures. In *Advances in Intelligent Systems and Computing Series, Proceedings of the Intelligent and Fuzzy Techniques in Big Data Analytics and Decision Making INFUS 2019,*; Springer: Cham, Switzerland, **2020**; 1029, pp. 133–140.

[6] Beliakov, G. On random generation of supermodular capacities. *IEEE Trans. Fuzzy Syst.* **2020**.

[7] Gutiérrez, I.; Gómez, D.; Castro, J.; Espínola, R. Fuzzy Measures: A solution to deal with community detection problems for networks with additional information. *J. Intell. Fuzzy Syst.* **2020**, *39*, 6217–6230.

[8] Blondel, V.; Guillaume, J.; Lambiotte, R.; Lefevre, E. Fast unfolding of communities in large networks. *J. Stat.-Mech. Theory Exp.* **2008**, *10*.

[1] https://github.com/inmaggp/Community-Detection-Problem-Based-on-Polarization-Measures.-An-application-to-Twitter-the-COVID-19-

Rough-Set-Driven Approach for Attribute Reduction in Fuzzy Formal Concept Analysis

1st M. José Benítez-Caballero
Departament of Mathematics
University of Cádiz
Cádiz, Spain
mariajose.benitez@uca.es

2nd Jesús Medina
Departament of Mathematics
University of Cádiz
Cádiz, Spain
jesus.medina@uca.es

3rd Eloísa Ramírez-Poussa
Departament of Mathematics
University of Cádiz
Cádiz, Spain
eloisa.medina@uca.es

4th Dominik Ślęzak
Institute of Informatics
University of Warsaw
Warsaw, Poland
slezak@mimuw.edu.pl

Abstract*—This work concerns the research recently published in *M. J. Benítez-Caballero, J. Medina, E. Ramírez-Poussa and D.Ślęzak, "Rough-Set-Driven Approach for Attribute Reduction in Fuzzy Formal Concept Analysis", Fuzzy Sets and Systems, vol. 391, pp. 117–138, 2020.

***Index Terms*—fuzzy sets, attribute reduction, reduct, formal concept analysis, rough set theory**

I. Introduction

The study of the knowledge stored in databases is one of the most important goals in several research fields. Formal Concept Analysis (FCA) [10] and Rough Set Theory (RST) [8] are two widely studied mathematical theories, devoted to obtain information from relational databases that contain uncertainty.

The reduction of size of database is a widely study issue in both theories, separately. In addition, in the literature, several papers can be found that stablish the existing connections between these two mathematical tools, considering the classical framework [3], [5], [7], [9].

This paper introduces a novel mechanism to reduce the set of attributes in the fuzzy general framework of multi-adjoint concept lattices, considering the RST philosophy with tolerance relations. This study extends the one introduced in [3] in a fuzzy case, shown that the same properties are not satisfied. Besides that, the proposed mechanism has been enriched with other interesting properties, showing that the new procedure also keeps important features. One of these properties is that the reduction is directly applied to the context and the whole concept lattice is not needed to be computed. Moreover, the main structure, based on the join-irreducible elements, is preserved including no new join-irreducible element after the reduction procedure. The notions and results obtained considering this framework is deeply studied in [4].

II. Reduction in Multi-adjoint Concept Lattices

In this section, we will present the proposed reduction mechanism to multi-adjoint concept lattices, which can also be applied to any other fuzzy FCA framework. For that, the RST reduction philosophy and a family of tolerance relations will be taken into account. In the following, we explain step by step how this reduction procedure is carried out.

Partially supported by the State Research Agency (AEI) and the European Regional Development Fund (FEDER) project TIN2016-76653-P.

Reduction in Multi-adjoint Concept Lattices from RST with tolerance relations

S1. The first step is to consider the associated information system (B, A), with the mappings $\bar{a}\colon B \to V_a$, from a given multi-adjoint formal context (A, B, R, σ) and a frame $(L_1, L_2, P, \&_1, \ldots, \&_n)$.

S2. From this context the sets V_a will be the poset P and the mappings $\bar{a}\colon B \to P$ will be defined as $\bar{a}(x) = R(a, x)$, for each $a \in A$ and $x \in B$.

S3. In this new environment, we consider the tolerance relations for each attribute, which are used to build the unidimensional $\mathcal{E}$-discernibility function of the associated information system.

S4. We compute the $\mathcal{E}$-information reducts $D_1, \ldots, D_n$.

S5. Considering these reducts we reduce the original fuzzy context, obtaining the reduced ones $(D_1, B, R_{|D_1 \times B}), \ldots, (D_n, B, R_{|D_n \times B})$.

S6. Finally, we build the concept lattices from the reduced contexts obtaining significant reductions in the context as well as in the size of the original lattice, preserving the discernibility among the objects.

From now on, a multi-adjoint frame $(L_1, L_2, P, \&_1, \ldots, \&_n)$ and a multi-adjoint context (A, B, R, σ) will be fixed. First of all, different properties relating the concept-forming operators in the original context and in the reduced one will be introduced in the following proposition, which will be needed later on.

Proposition 1: Given a subset $D \subseteq A$, for any concept $\langle g, f\rangle$ obtained from the multi-adjoint context, the following statements hold:

1) $g^{\uparrow\downarrow} \preceq_2 g^{\uparrow_D\downarrow^D}$
2) $g^{\uparrow\downarrow\uparrow_D} = g^{\uparrow_D}$
3) $f^{\downarrow^D\uparrow\downarrow} = f^{\downarrow^D}$

The following proposition asserts that if a group of concepts in the original context are the same concept in the reduced one, then they have the structure of a join-semilattice.

Proposition 2: Let $D \subseteq A$ be a subset of attributes. The set $R_E = \{(\langle g_1, f_1\rangle, \langle g_2, f_2\rangle) \mid \langle g_1, f_1\rangle, \langle g_2, f_2\rangle \in$

$\mathcal{M}(A,B,R,\sigma), g_1^{\uparrow_D\downarrow^D} = g_2^{\uparrow_D\downarrow^D}\}$ is an equivalence relation and every class $[\langle g,f\rangle]_D$ of $\mathcal{M}(A,B,R,\sigma)/R_E$ is a join-semilattice with maximum element $\langle g^{\uparrow_D\downarrow^D}, g^{\uparrow_D\downarrow^D\uparrow}\rangle$.

We can think that it is natural that properties presented in [3] will not be satisfied when fuzzy sets are considered. Concerning the structural properties, one of the most important feature is related to the join-irreducible elements. As it was proven in Theorem 3.5 of [3] for the classical framework of FCA, no new join-irreducible element appears after the reduction process. This main structural property is also preserved in this fuzzy framework, as the following result shows.

Theorem 3: Given an $\mathcal{E}$-information reduct D in the corresponding context information system (B,A), if the pair $\langle g,f\rangle$ is a join-irreducible concept in the reduced concept lattice built using the reduct D, then there exists an object $b\in B$ and a truth value $x\in L_2$ such that $\phi_{b,x}^{\uparrow_D}=f$ and $\phi_{b,x}$ also generates a join-irreducible concept in the original concept lattice.

III. Related methodologies

The attribute reduction mechanism based on tolerance relations proposed in this paper is different from the ones given in diverse papers. In this section, we will focus our attention on two general reduction procedures given in [1] and [6], since the rest of procedures are given in a more restrictive framework or are based on them.

The philosophy of the reduction presented in [1], considering similarities, is very different from the proposed one in this paper. Our reduction is directly applied to the context. This fact is very important since, for example, it has a direct impact on the construction of the attribute implications, such as, reducing the number of implications or creating equivalences among them. Meanwhile, the factorization presented by Bělohlávek reduces the concept lattice and it has no impact in the context. Moreover, the whole concept lattice must be computed before calculating the factorization, which is not necessary for the mechanism we propose. Furthermore, the reduction given in [1] provides a covering instead of a partition, which is the clustering we obtain from the $\frac{1}{2}$-information reducts.

On the other hand, the reduction presented in [6] consideres block relations, where the rows of this relation are intents and the columns are extents. The methodology proposed in this paper is different, since the tolerance relations are defined on the set of values of the attributes, and they are independent of the relation of the context. With our procedure, we try to group the similar attributes whose separated consideration does not provide relevant information and so, the main knowledge of the database is preserved after the reduction. Although both mechanisms are different, they are compatible. As a consequence, we can apply both procedures to the same database in order to obtain a bigger reduction, embedding both philosophies.

IV. Conclusions and future work

In this paper, by means of different results and examples, we have introduced a mechanism to reduce attributes in fuzzy FCA, considering the reduction procedure with tolerance relations introduced in RST [2]. This new method to reduce attributes provides a significant reduction of the original concept lattice. Some interesting properties of the new procedure have been presented troughout the paper. The most important one shows that the structure of the original concept lattice is partially preserved considering this new mechanism, that is, no new join-irreducible elements appear after carrying out the reduction procedure.

Moreover, the paper finishes exposing a comparison among this new mechanism and other interesting fuzzy reduction methods.

In the future, we are interested in establishing a comparison between the attribute reduction process given in multi-adjoint concept lattices and other multi-adjoint frameworks, as multi-adjoint property oriented concept lattice and multi-adjoint object oriented concept lattices. We also want to apply this study to real examples.

References

[1] R. Bělohlávek. Similarity relations in concept lattices. *Journal of Logic and Computation*, 10(6):823–845, 2000.

[2] M. J. Benítez-Caballero, J. Medina, E. Ramírez-Poussa, and D. Ślęzak. Bireducts with tolerance relations. *Information Sciences*, 435:26 – 39, 2018.

[3] M. J. Benítez-Caballero, J. Medina, E. Ramírez-Poussa, and D. Ślęzak. A computational procedure for variable selection preserving different initial conditions. *International Journal of Computer Mathematics*, 2019.

[4] M. J. Benítez-Caballero, J. Medina, E. Ramírez-Poussa, and D. Ślęzak. Rough-set-driven approach for attribute reduction in fuzzy formal concept analysis. *Fuzzy Sets and Systems*, 2019.

[5] J. Chen, J. Li, Y. Lin, G. Lin, and Z. Ma. Relations of reduction between covering generalized rough sets and concept lattices. *Information Sciences*, 304:16–27, 2015.

[6] J. Konecny and M. Krupka. Complete relations on fuzzy complete lattices. *Fuzzy Sets and Systems*, 320:64 – 80, 2017. Theme Logic and Algebra.

[7] J. Medina. Relating attribute reduction in formal, object-oriented and property-oriented concept lattices. *Computers & Mathematics with Applications*, 64(6):1992–2002, 2012.

[8] Z. Pawlak. Rough sets. *International Journal of Computer and Information Science*, 11:341–356, 1982.

[9] L. Wei and J.-J. Qi. Relation between concept lattice reduction and rough set reduction. *Knowledge-Based Systems*, 23(8):934–938, 2010.

[10] R. Wille. Restructuring lattice theory: an approach based on hierarchies of concepts. In I. Rival, editor, *Ordered Sets*, pages 445–470. Reidel, 1982.

Measuring Consistency of Fuzzy Logic Theories

Nicolás Madrid
Departamento de Matemática Aplicada
Universidad de Málaga
Málaga, Spain
Email: nicolas.madrid@uma.es

Manuel Ojeda-Aciego
Departamento de Matemática Aplicada
Universidad de Málaga
Málaga, Spain
Email: aciego@uma.es

***Abstract*—Fuzzy logic has shown to be a suitable framework to handle contradictions in which, unsurprisingly, the notion of inconsistency can be defined in different ways. This paper starts with a short survey of different ways to define the notion of inconsistency in fuzzy logic systems. As a result, we provide a first notion of inconsistency by means of the absence of models. Subsequently, we define two measures of consistency that belong purely to the fuzzy paradigm; in the sense that both measures coincide with the crisp notion of consistency when the set of truth values is $\{0, 1\}$. Accordingly, we can state that the two provided measures of consistence are notions of consistence based on degrees, bringing back the spirit of fuzzy logic into the notion of consistency.**

Based on the paper *A measure of consistency for fuzzy logic theories*, to appear in the journal *Mathematical Methods in the Applied Sciences*, 2021.

I. Introduction

Since its introduction, fuzzy sets and fuzzy logic have shown to be an interesting research topic. One can find lots of papers ranging from the development of algebraic theories of fuzzy structures or the underlying mathematics of fuzzy logic, to fuzzy modelling or automated control in terms of sets of fuzzy rules. From the theoretical standpoint; in [1] the notion of relational Galois connection is extended to be applied between transitive fuzzy directed graphs in a framework in which the components of the connection are crisp relations satisfying certain reasonable properties; from the practical standpoint, in [7] it is shown how a control application can leverage (even) from a set of inconsistent rules; and in [2] we can see a fuzzy logic-based mathematical model of a sequence of earthquakes using tools from fuzzy reasoning.

Being the fuzzy realm a matter of degrees, a number of papers have focused on measuring the degree of inconsistency of a set of fuzzy rules, and a number of different inconsistency indices have been introduced. For instance [3] introduces the so-called knowledge-based consistency index for deriving priorities from fuzzy pairwise comparison matrices in multiple-criteria decision-making problems; other approaches introduce means for both measuring and repairing inconsistency, for example [8] presents a family of measures aimed at determining the amount of inconsistency in knowledge bases with graded truth and considers minimal adjustments in the truth-degrees of the propositions necessary to make the knowledge-base to be consistent within a given frame (in that case the Łukasiewicz semantics); last but not least [4] deals with the definition of measures of inconsistency in the residuated-logic-programming paradigm under the fuzzy answer set semantics and provides a soft mechanism to control the amount of information inferred, thus, controlling the inconsistencies by modifying slightly the truth values of some rules. The number of possible measures of inconsistency that can be found in the literature somehow suggests the existence of a problem with inconsistency in a fuzzy setting, namely, its definition: there is not a consensus on how to interpret inconsistency in a fuzzy system.

In this paper we briefly survey the main properties and equivalent characterisations of inconsistency in classical (crisp) logic and then, we focus on, under of point of view, the more natural way to define inconsistency in a logic theory, namely: the absence of models. This consideration as definition of inconsistency keeps some of the most important properties of inconsistency in the fuzzy paradigm, e.g., explosive reasoning. However, we also lose an important issue, we lose degrees; which is the soul of fuzzy logic. For such a reason, we propose a generalization of consistence by means of two measures of consistency. Specifically, we define two measures of consistency that belong purely to the fuzzy paradigm. In other words, both measures coincide with the crisp notion of consistency when the set of truth values is $\{0, 1\}$. Moreover, we provide a set of properties for both measures of consistency in order to motivate the use of them to represent the consistency of fuzzy logic theories.

II. Measures of consistency for fuzzy logic theories.

In this section we follow a common procedure in the definition of measures of (in-)consistency in crisp logic: given a logic theory Γ, we consider subsets of consistent formulas contained in Γ. At this point, in crisp logic we can measure the inconsistency by considering the ratio or the absolute number of removed formulas. Interestingly enough, in a fuzzy environment, we can proceed differently: for instance, we can measure the consistency Mc of the removed formulas with respect to the remaining ones.

Definition 1: Let Γ be a fuzzy logic theory defined on a residuated lattice $(L, \leq, *, \rightarrow)$ and consider $\alpha \in L$. A formula ψ is said to be α-*feasible* w.r.t. Γ if $\Gamma \models \psi \rightarrow \alpha$. If $\Gamma = \emptyset$

The *degree of consistency of* ψ *with respect to* Γ is:

$$Mc(\psi, \Gamma) = \min\{\alpha \mid \psi \text{ is } \alpha\text{-feasible w.r.t } \Gamma\}$$

In order to properly understand the rationale in the following definition, let us consider an arbitrary fuzzy logic theory of three formulas $\Gamma = \{\psi_1, \psi_2, \psi_3\}$. Assume that Γ is inconsistent and that ψ_1 is non-contradictory. The question is how consistent are formulas ψ_2 and ψ_3 in the logic theory $\Gamma^* = \{\psi_1\}$. One could think about measuring separately the consistence for both formulas and then aggregate them, but that is not possible because $\{\psi_1, \psi_2\}$ and $\{\psi_1, \psi_3\}$ could be consistent and, then, both measures would be 1. Therefore, the only reasonable option is to combine ψ_2 and ψ_3 into one formula. Note, that the consistency of $\{\psi_1, \psi_2, \psi_3\}$ is given by assuming on the one hand ψ_1 and, on the other hand, *both* ψ_2 *and* ψ_3 at the same time; the latter means that we are assuming $\psi_2 \wedge \psi_3$. Therefore, the consistency generated by ψ_2 and ψ_3 in the logic theory $\Gamma^* = \{\psi_1\}$ is $Mc(\psi_2 \wedge \psi_3, \Gamma^*)$.

Definition 2: Let Γ be a fuzzy logic theory defined on a residuated lattice $(L, \leq, *, \rightarrow)$, then we define the measure of consistency $Mc^*(\Gamma)$ as

$$\sup \left\{ Mc\left(\bigwedge_{\psi_i \in \Gamma \setminus \Gamma^*} \psi_i \ , \Gamma^* \right) \mid \Gamma^* \subseteq \Gamma \text{ is consistent} \right\}.$$

At first sight, the reader may think that considering all the set of combinations of consistent sub-theories of a fuzzy logic theory Γ may be unpractical, however, the following result shows that only one consistent subtheory must be considered to compute the measure Mc^*, namely, the empty theory.

Theorem 1: Let Γ be a fuzzy logic theory defined on a residuated lattice $(L, \leq, *, \rightarrow)$, then:

$$Mc^*(\Gamma) = Mc\left(\bigwedge_{\psi_i \in \Gamma} \psi_i \ , \emptyset \right).$$

As a direct consequence of the previous theorem, we have the following corollary that shows that Mc^* satisfies those properties of a measure of consistency (i.e., the opposite properties of a measure of inconsistency).

Corollary 1: Let Γ and Γ' be fuzzy logic theories defined on a residuated lattice $(L, \leq, *, \rightarrow)$, then:

a) $Mc^*(\Gamma) \geq Mc^*(\Gamma \cup \Gamma')$;
b) If Γ is consistent then, $Mc^*(\Gamma) = 1$;
c) If $Mc^*(\Gamma) \neq 1$ then, Γ is inconsistent;
d) If L is finite and totally ordered, then $Mc^*(\Gamma) = 1$ implies Γ is consistent.

The measure of consistency Mc^* is related to the k-models which, in turn, are related to the so-called x-consistency [9] and α-cuts models [6]. The underlying idea in the k-models is to guarantee the satisfiability of formulas in at least truth-degree $k \in L$, and it is given in the following definition.

Definition 3: Let Γ be a fuzzy logic theory defined on a residuated lattice $(L, \leq, *, \rightarrow)$ and consider $k \in L$. We say that an interpretation M is a k-model of Γ if $M(\psi) \geq k$ for all $\psi \in \Gamma$.

The k-models were introduced in the context of Fuzzy Logic Programming aiming at providing "partial" models to a given inconsistent logic program (i.e., fuzzy logic theory). Later, it was proved that the existence of models is guaranteed by very general requirements in Fuzzy Logic Programming [5] and, then, k-models faded away. However, in the general context we are working on in this approach, the existence of models cannot be guaranteed easily and k-models may be valuable here. The following result relates the measure Mc^* with k-models.

Theorem 2: Let Γ be a fuzzy logic theory defined on a residuated lattice $(L, \leq, *, \rightarrow)$. If $Mc^*(\Gamma) = \alpha$ then, there is not β-model of Γ with $\beta > \alpha$.

III. Conclusions and Future Work

We have presented two different measures of consistency. The first one measures how much compatible a formula is with respect to a given theory in the sense *the closer to* 0*, the more inconsistent; and the closer to* 1*, the more consistent*. The second measure determines a degree of consistency of a logic theory by means of consistent subtheories. Both definitions coincide with the standard notion of consistency when we restrict to crisp logic, and both definitions satisfy convenient properties in order to be considered measures of consistency.

There are two main lines of future research. On the one hand it is convenient to keep digging up some measures of inconsistency in fuzzy paradigms. To have a notion of inconsistency based on degrees (as the ones proposed in this paper) may allow to incorporate a paraconsistent reasoning into inconsistent fuzzy logic theories without leaving the fuzzy paradigm aside. On the other hand, it is interesting to find out an application of the measures of consistence. For instance, we think they can be used to deal with contradictions in databases obtained from fails or system errors.

Acknowledgment

Partially supported by projects PGC2018-095869-B-I00 and UMA2018-FEDERJA-001.

References

[1] I. Cabrera, P. Cordero, E. Muñoz-Velasco, M. Ojeda-Aciego, and B. De Baets. Relational galois connections between transitive fuzzy digraphs. *Mathematical Methods in the Applied Sciences*, 43(9):5673–5680, 2020.

[2] A. Konguetsof, N. Mylonas, and B. Papadopoulos. Fuzzy reasoning in the investigation of seismic behavior. *Mathematical Methods in the Applied Sciences*, 43(13):7747–7757, 2020.

[3] S. Kubler, W. Derigent, A. Voisin, J. Robert, Y. L. Traon, and E. Herrera-Viedma. Measuring inconsistency and deriving priorities from fuzzy pairwise comparison matrices using the knowledge-based consistency index. *Knowl. Based Syst.*, 162:147–160, 2018.

[4] N. Madrid and M. Ojeda-Aciego. Measuring inconsistency in fuzzy answer set semantics. *IEEE Trans. Fuzzy Syst.*, 19(4):605–622, 2011.

[5] N. Madrid and M. Ojeda-Aciego. On the existence and unicity of stable models in normal residuated logic programs. *Intl J of Computer Mathematics*, 89(3):310–324, 2012.

[6] J. Močkoř. α-cuts and models of fuzzy logic. *International Journal of General Systems*, 42(1):67–78, 2013.

[7] N. R. Pal and K. Pal. Handling of inconsistent rules with an extended model of fuzzy reasoning. *J. Intell. Fuzzy Syst.*, 7(1):55–73, 1999.

[8] D. Picado-Muiño. Measuring and repairing inconsistency in knowledge bases with graded truth. *Fuzzy Sets Syst.*, 197:108–122, 2012.

[9] D. Van Nieuwenborgh, M. De Cock, and D. Vermeir. An introduction to fuzzy answer set programming. *Annals of Mathematics and Artificial Intelligence*, 50(3):363–388, 2007.

Detecting equivalence classes being sublattices from an attribute reduction in FCA

Roberto G. Aragón*
Department of Mathematics
University of Cádiz
Cádiz, Spain
roberto.aragon@uca.es

Jesús Medina
Department of Mathematics
University of Cádiz
Cádiz, Spain
jesus.medina@uca.es

Eloísa Ramírez-Poussa
Department of Mathematics
University of Cádiz
Cádiz, Spain
eloisa.ramirez@uca.es

Abstract*—This work concerns the research recently published in *R. G. Aragón, J. Medina, and E. Ramírez-Poussa. Identifying Non-Sublattice Equivalence Classes Induced by an Attribute Reduction in FCA. Mathematics, 9(5), 2021.

***Index Terms*—Formal Concept Analysis, equivalence relations, attribute reduction**

I. Introduction

An appealing goal in different frameworks is detecting redundant or irrelevant variables (attributes) in data sets, such as in Formal Concept Analysis (FCA) in which the removal of redundant data becomes essential. However, the elimination of such variables may have some impact in the concept lattice, which is closely related to the algebraic structure of the obtained quotient set and their classes [4].

In [2], [3], local congruences were introduced as equivalence relations defined on lattices whose equivalence classes are sublattices of the original lattice. Furthermore, local congruences were intended to complement the attribute reductions of formal contexts in order to ensure that the equivalence classes, $[C]_D$, be sublattices of the original concept lattice. If its infimum $C_m = \bigwedge_{C_i \in [C]_D} C_i$ belongs to the equivalence class, we can assert that the class is already a sublattice, since a join-semilattice with a minimum element is a lattice. As a consequence, in this case, the application of a local congruence, as a complementary mechanism to attribute reduction, does not provide any modification in this particular class. Therefore, it is significant to characterize the required conditions under which these cases arise.

II. Characterizing equivalence classes

This research line was initiated in [2] and was continued in [1]. Namely, we determined in this last paper a sufficient condition to ensure that the equivalence class of C_m is generated by an attribute-concept and presented the following enhanced version of the characterization of the infimum of elements belonging to a non-singleton classes:

Theorem 1: Given a context (A, B, R), a subset of attributes $D \subseteq A$, and a concept $C \in \mathcal{C}(A, B, R)$ such as its equivalence class $[C]_D$ of the induced equivalence relation is not a singleton. We have that $C_m \notin [C]_D$ if and only if one of the following statements is satisfied:

- There exists at least one attribute $a \in D$ such that $C_m = (a^{\downarrow}, a^{\downarrow\uparrow})$.
- There exists a concept $C^* \in \mathcal{C}(A, B, R)$, such as $C^* = (a^{*\downarrow}, a^{*\downarrow\uparrow})$ with $a^* \in D$, $C^* \notin [C]_D$ and $C_M \not\leq C^*$. Moreover, $\overline{C^*}$ is in a meet-irreducible decomposition $\{\overline{C_j} \in M_F(D, B, R_{|D\times B}) \mid j \in J\}$ of $\overline{C_m}$.

In addition, we analyzed Theorem 1 when the considered subset of attributes does not contain unnecessary attributes. This fact is the usual case in FCA attribute reduction and simplifies the detection of equivalence classes which are not convex sublattices of the original concept lattice. Furthermore, under this consideration, we also proved that when the original concept lattice is isomorphic to a distributive lattice the induced equivalence classes by the reduction are always sublattices.

III. Conclusions

Consequently, all the results shown in this paper have a significant importance, for example, in the application of local congruences, due to characterizing the cases when classes are not sublattices, we will be useful to know which classes will be affected when a local congruence be applied after an attribute reduction mechanism.

* Corresponding author

Partially supported by the 2014–2020 ERDF Operational Programme in collaboration with the State Research Agency (AEI) in project PID2019-108991GB-I00, and with the Department of Economy, Knowledge, Business and University of the Regional Government of Andalusia in project FEDER-UCA18-108612, and by the European Cooperation in Science & Technology (COST) Action CA17124.

References

[1] R. G. Aragón, J. Medina, and E. Ramírez-Poussa. Identifying non-sublattice equivalence classes induced by an attribute reduction in fca. *Mathematics*, 9(5), 2021.

[2] R. G. Aragón, J. Medina, and E. Ramírez-Poussa. Impact of local congruences in attribute reduction. In M.-J. Lesot, S. Vieira, M. Z. Reformat, J. P. Carvalho, A. Wilbik, B. Bouchon-Meunier, and R. R. Yager, editors, *Information Processing and Management of Uncertainty in Knowledge-Based Systems*, volume 1239, pages 748–758, Cham, 2020. Springer International Publishing.

[3] R. G. Aragón, J. Medina, and E. Ramírez-Poussa. Reducing concept lattices by means of a weaker notion of congruence. *Fuzzy Sets and Systems*, 2020. https://doi.org/10.1016/j.fss.2020.09.013.

[4] M. J. Benítez-Caballero, J. Medina, E. Ramírez-Poussa, and D. Ślęzak. A computational procedure for variable selection preserving different initial conditions. *International Journal of Computer Mathematics*, 97(1-2):387–404, 2020.

Non-monotonic multi-adjoint logic programming with constraints

M. Eugenia Cornejo
Department of Mathematics
University of Cádiz
Cádiz, Spain
mariaeugenia.cornejo@uca.es

David Lobo*
Department of Mathematics
University of Cádiz
Cádiz, Spain
david.lobo@uca.es

Jesús Medina
Department of Mathematics
University of Cádiz
Cádiz, Spain
jesus.medina@uca.es

***Abstract*—This work concerns the research recently published in *M. E. Cornejo, D. Lobo and J. Medina, "Extended multi-adjoint logic programming", Fuzzy Sets and Systems, vol. 388, pp. 124–145, 2020.* The reader is referred to that source for a full discussion and examples of the work.**

***Index Terms*—multi-adjoint logic programming, non-monotonic logic programming, negation operator, stable models**

I. Introduction

Medina et al. introduced multi-adjoint logic programming [8] in order to generalize different non-classical logic programming frameworks [4], [9]. The foundations of a multi-adjoint logic program is a complete lattice endowed with different adjoint pairs, which allow to obtain several generalizations of modus ponens. Recently, the inclusion of a negation operator in multi-adjoint logic programming was carried out in [1], giving rise to a first non-monotonic multi-adjoint logic approach, which generalizes other current frameworks [6], [7].

In [2], we defined a general non-monotonic logic programming language which shares the multi-adjoint philosophy. Besides permitting different adjoint pairs, the syntax of the so-called extended multi-adjoint logic programming is characterized by the use of constraints and of a special type of aggregator operator. Namely, such aggregation enables to consider, among others, different negation operators in the body of the same rule of the logic program. In addition to defining the syntax and the semantics of this logic programming paradigm, a mechanism for obtaining a multi-adjoint normal logic program from an extended multi-adjoint logic program has been shown, which entails a correlation between the semantics of both logic programming languages.

II. Extended multi-adjoint logic programming

The underlying idea of extended multi-adjoint logic programs can be stated as follows:

- On the one hand, constraints simulate the limitation of a certain property, attribute or characteristic by an upper bound. Therefore, the inclusion of constraint rules of the form $\langle c \leftarrow_i \mathcal{B};\ \top\rangle$ in a logic programming language might be enormously useful in what regards applications, being c an element of a lattice representing a degree of limitation.
- On the other hand, as well as the multi-adjoint paradigm aims at relaxing limitations for modus ponens, it would be desirable to provide freedom for the non-monotonic behaviour of propositional symbols. For instance, it could be advantageous allowing different negations in the body of the same rule. This can be done by considering *extended aggregators*, that is, an n-ary mapping $@^e$ defined as:

  $$@^e(x_1,\ldots,x_n) = @^e[x_1,\ldots,x_m;x_{m+1},\ldots,x_n]$$

  such that it is order-preserving in the first m arguments and order-reversing in the last $n-m$ arguments.

According to the foregoing remarks, the syntax of extended multi-adjoint logic programming is defined as follows.

Definition 1: Let $(L,\preceq,\leftarrow_1,\&_1,\ldots,\leftarrow_n,\&_n,@_1^e,\ldots,@_k^e)$ be an extended multi-adjoint lattice with greatest element $\top$. An *extended multi-adjoint logic program* is a finite set of weighted rules of the form

$$\langle p \leftarrow_i @^e[p_1,\ldots,p_m;p_{m+1},\ldots,p_n];\vartheta\rangle$$

and constraint rules of the form

$$\langle c \leftarrow_i @^e[p_1,\ldots,p_m;p_{m+1},\ldots,p_n];\top\rangle$$

where $i\in\{1,\ldots,n\}$, $@^e\in\{@_1^e,\ldots,@_k^e\}$, $\vartheta,c\in L$ and $p_{s_1}\neq p_{s_2}$, for all $s_1,s_2\in\{1,\ldots,n\}$ with $s_1\neq s_2$.

The semantics of extended multi-adjoint logic programs is defined similarly to the stable model semantics of multi-adjoint normal logic programs [1]. Namely, the stable model semantics lies on the principles of the Gelfond-Lifschitz reduct [5].

A detailed procedure to transform an extended multi-adjoint logic program into a semantically equivalent multi-adjoint normal logic program was introduced in [2]. Such a procedure is shown in two steps:

1) Firstly, given an extended multi-adjoint normal logic program, its constraint rules are converted into regular

* Corresponding author

Partially supported by the 2014–2020 ERDF Operational Programme in collaboration with the State Research Agency (AEI) in project PID2019-108991GB-I00, and with the Department of Economy, Knowledge, Business and University of the Regional Government of Andalusia in project FEDER-UCA18-108612, and by the European Cooperation in Science & Technology (COST) Action CA17124.

rules, in such a way that the stable models of the original and the final logic programs coincide.

2) Then, the remaining rules are written in terms of a single non-monotonic unary mapping, which turns out to be an involutive negation. It needs to be stressed that such transformation is carried out by means of continuous mappings. An interesting outcome of this fact is the possibility to apply a sufficient condition for the existence of stable models in multi-adjoint normal logic programs [1] for constraint-free extended multi-adjoint logic programs.

III. Conclusions

The research carried out in [2] enables to make use of a flexible language like extended multi-adjoint logic programs in order to model real-world problems, and then translate them into multi-adjoint normal logic programs to handle compact simple programs with the same meaning. Furthermore, this procedure can be complemented with the methods shown in [3], where a multi-adjoint normal logic program is translated into a core fuzzy answer set program. As highlighted in [6], core fuzzy answer set programs are easier to implement and to reason about from a computational point of view.

The completed transformation considerably increases the potential of extended multi-adjoint logic programs to model real-life problems, since modelling the information contained in a text or in a database by decision rules and the interpretation of those rules will be easier through extended multi-adjoint logic programs, and its translation into a core fuzzy answer set program will facilitate the simulation and computation of the consequences/deductions from the program.

References

[1] M. E. Cornejo, D. Lobo and J. Medina, “Syntax and semantics of multi-adjoint normal logic programming”, Fuzzy Sets and Systems, vol. 345, pp. 41–62, 2018.

[2] M. E. Cornejo, D. Lobo and J. Medina, “Extended multi-adjoint logic programming”, Fuzzy Sets and Systems, vol. 388, pp. 124–145, 2020.

[3] M. E. Cornejo, D. Lobo and J. Medina, “Relating multi-adjoint normal logic programs to core fuzzy answer set programs from a semantical approach”, Mathematics, vol. 8 (6), 881, 2020.

[4] C.V. Damásio, L. M. Pereira, “Monotonic and residuated logic programs”, in Symbolic and Quantitative Approaches to Reasoning with Uncer-tainty (ECSQARU’01), in Lecture Notes in Artificial Intelligence, vol. 2143, pp. 748–759, 2001.

[5] M. Gelfond and V. Lifschitz, “The stable model semantics for logic programming”, in Logic Programming: Proceedings of the Fifth International Conference and Symposium, MIT Press, pp. 1070–1080, 1988.

[6] J. Janssen, S. Schockaert, D. Vermeir and M. De Cock, “A core language for fuzzy answer set programming”, International Journal of Approximate Reasoning, vol. 53 (4), pp. 660–692, 2012.

[7] N. Madrid and M. Ojeda-Aciego, “On the existence and unicity of stable models in normal residuated logic programs”, International Journal of Computer Mathematics, vol. 89 (3), pp. 310–324, 2012.

[8] J. Medina, M. Ojeda-Aciego and P. Vojtáš, “Multi-adjoint logic programming with continuous semantics” in Logic Programming and Non-Monotonic Reasoning (LPNMR’01), vol. 2173, pp. 351–364, 2001.

[9] P. Vojtáš, “Fuzzy logic programming”, Fuzzy Sets and Systems, vol. 124 (3), pp. 361–370, 2001.

XIV Congreso Español de Metaheurísticas, Algoritmos Evolutivos y Bioinspirados



XIV Congreso Español de Metaheurísticas, Algoritmos Evolutivos y Bioinspirados

Sesión General



On the role of the gradient optimizer on evolved GANs

Unai Garciarena
Intelligent Systems Group
University of the Basque Country, UPV/EHU
Donostia-San Sebastian, Spain
unai.garciarena@ehu.eus

Alexander Mendiburu
Intelligent Systems Group
University of the Basque Country, UPV/EHU
Donostia-San Sebastian, Spain
alexander.mendiburu@ehu.eus

Roberto Santana
Intelligent Systems Group
University of the Basque Country, UPV/EHU
Donostia-San Sebastian, Spain
roberto.santana@ehu.eus

***Abstract*—Recently, different neuroevolutionary approaches have been proposed to enhance the structure (architecture and hyperparameters) of generative neural models. In this paper, we investigate the role played by the gradient-based optimizer chosen to train a generative adversarial net (GAN) when its architecture has been obtained using neuroevolution. We focus on 2D Gaussian mixture approximation problem and evaluate the effect of a set of representative gradient-based techniques on the quality of the performance of the GANs. Our results show that the particular choice of the gradient optimizer can be as relevant as the appropriate selection of the architecture.**

***Index Terms*—GANs, generative models, hybrid methods, gradient-based optimization, neuroevolution**

I. Introduction

In the last few years, several efforts for automatically designing the architecture of deep neural networks (DNN) have been devoted, particularly using EAs [1]–[3]. While most of these works focus of models comprising a single neural network (e.g., multi-layer perceptron (MLP), convolutional neural network, etc.), some recent research [4]–[8] investigate multi-component networks, such as the generative adversarial networks (GAN) [9]. These models are mainly used as generators of new data and in this case, finding the appropriate architecture for each component is challenging.

Furthermore, in most of the neuroevolutionary approaches for deep neural networks, the object of evolution is the model architecture and the set of hyperparameters of the network, e.g., the activation functions, or the layer/kernel size. Sometimes, the weights themselves are also included in the evolutionary procedure, whereas other aspects as important (if not more important) hardly ever are. Components such as the loss function, or the stochastic gradient descent [10]–[12] (SGD) method (which also are part of the model structure), critical to the correct training of DNN models, are commonly ignored in neural architecture search algorithms.

One of the reasons that could explain why the consideration of some of these aspects has been neglected is the lack of a clear understanding on whether and how, elements such as the choice of the gradient optimizer may have a significant impact in the behavior of a given model. A common assumption is to expect that if the structure of DNN is robust, then the choice of the gradient optimizer will not make a significant impact in the performance of the network. We test this assumption by using a number of evolved structures as a baseline and evaluating their performance when trained using six gradient optimizers (and combinations of these) which correspond to different search strategies. We also investigate the question of whether the gradient optimizer used in each component of a multi-network model should be the same or different. We focus on GANs that use as its main components MLPs.

The paper is organized as follows: The next section presents some necessary background on GANs and reviews related works. Section III presents a brief description of the method used to evolve the GAN structures and a summary of previous findings on the role of some of the GAN components in its behavior, as well as unresolved questions in that same matter. Section IV presents the experimental framework used to compare the different variants of gradient optimizers, and discusses the results obtained within this framework. Finally, the conclusions of the paper and some lines for further research are presented in Section V.

II. Background

A. Generative adversarial networks

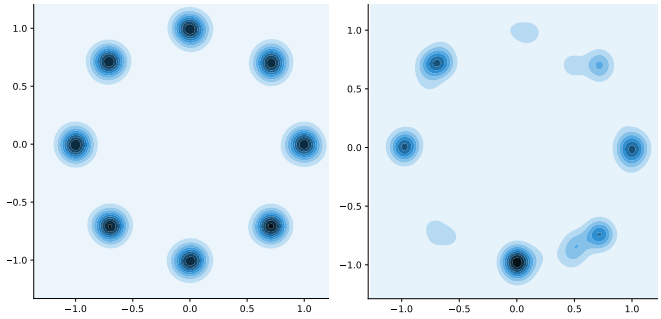
A GAN [9] model, as following its original description, is composed of two networks; a generator G and a discriminator D. The goal of the generator is to create samples that look as similar as possible to examples from the available data, in an effort to fool the discriminator. At the same time, the discriminator tries to correctly discern the examples found in the original data from the samples produced by the generator.

The weights and biases of the generator are denoted by θ_g, and the generator's distribution over data $\mathbf{X}$ as p_g. $\mathbf{z}$ stands for the input random noise sampled from a latent variable that G receives. This leads to the generated samples, $\hat{\mathbf{x}}$. Formally, $G(\mathbf{z}, \theta_g) \rightarrow \hat{\mathbf{x}}$.

Similarly, the discriminator parameters are represented with θ_d. The discriminator can receive as input true $\mathbf{x}$ or fake samples $\hat{\mathbf{x}}$ and it outputs probability values indicating how likely the samples are to be observations of the original data.

Equation 1 captures these two opposing goals:

$$\arg\min_{\theta^G}\max_{\theta^D} V(D,G) = \mathbb{E}_{\mathbf{x}\sim p_{data}(\mathbf{x})}[log(D(\mathbf{x},\theta^D))] + \mathbb{E}_{\mathbf{z}\sim p_{\mathbf{z}}(\mathbf{z})}[log(1 - D(G(\mathbf{z},\theta^{\mathbf{G}}),\theta^D))] \quad (1)$$



(a) Target distribution (b) Undesired effect.

Fig. 1: 8-mode, 2D Gaussian mixture problem target and approximation examples.

where $\mathbb{E}_{\mathbf{x}\sim p_{data}(\mathbf{x})}$ and $\mathbb{E}_{\mathbf{z}\sim p_{\mathbf{z}}(\mathbf{z})}$ represent, respectively, the expectations with respect to the original data distribution p_{data} and the latent variable distribution $p_{\mathbf{z}}(\mathbf{z})$ [9]. This equation, as it pursues the minimization and maximization of the same expression, can be translated into two loss functions which can be optimized.

A successful training of the model would result in the *fake* samples created by the generator resembling real ones.

The distribution of real-world data is usually within a multimodal space, where regions with a higher probability than others exist. An ideal generator should perfectly reproduce this distribution, avoiding the omission of areas with lower probability. However, GANs tend to be unsuccessful in this matter [13]. This issue, known as mode collapsing, is a drawback often held against GANs.

B. Gaussian Mixture approximation problem

The Gaussian mixture approximation problem [14] consists of generating a set of points x from a mixture of Gaussian distributions $x \sim \mathcal{N}(\boldsymbol{\mu}, \sigma)$, where $\boldsymbol{\mu} = \{\mu_0, \mu_1, ...\mu_{m-1}\}$ is a vector of means and σ is the unique variance parameter. This problem is especially designed to detect mode collapsing GANs, as models suffering from this deficiency easily learn to generate points from a subset of the m modes. In this case, the chosen specification is the 2D 8-mode variant; in which $m = 8$, and $\mu_i = (\mu_i^0, \mu_i^1)$. The two components of each μ_n can be interpreted as coordinates in a 2D grid. A visualization of the distribution of x is found in Figure 1a. It shows the 8 distribution centers (with $\sigma = 0.05$) describing a circle: $\mu_0 = (0, -1)$, $\mu_1 = (-\sqrt{2}, -\sqrt{2})$, $\mu_2 = (-1, 0)$, $\mu_3 = (-\sqrt{2}, \sqrt{2})$, $\mu_4 = (\sqrt{2}, -\sqrt{2})$, $\mu_5 = (1, 0)$, $\mu_6 = (\sqrt{2}, \sqrt{2})$, $\mu_7 = (0, 1)$.

After being trained, a generative model is sampled aiming to reproduce the target distribution. However, not all models are able to do so, producing results as shown in Figure 1b. In this example, a very reduced number of points have been generated where two distribution modes were placed, and some other points far from the original means have also been sampled.

In order to numerically test the quality of a set of samples (and thus, the quality of the generative model), the Maximum Mean Discrepancy[1] [15] (MMD) has been proposed as a fitting metric to produce a numeric value representing the difference between two distributions [5].

C. Related evolutionary approaches and transfer learning

In [5], an evolutionary approach, E-GAN, is proposed. It contemplates a single discriminator during the whole evolutionary procedure as the environment the generators need to adapt to. The mentioned work addresses the question of considering different loss functions for training the networks as a part of the evolutionary algorithm.

In [16], the authors propose Lipizzaner, an approach based on the diversity enhancing method of spatial evolution [17]. This particular method relies exclusively on the Wasserstein loss function, and describes two different mutation operators; a gradient based one (applying an iteration of learning using a minibatch), and a gradient-free option (modifying the learning rate).

The two mentioned works, E-GAN and Lipizzaner, are combined in [7] to form Mustangs, adding the three different mutation operators used in E-GAN to Lipizzaner. In this case, MLP networks are also considered (for evolving GANs that deal with the MNIST [18] problem).

Another approach which explores the convenience of evolving gradient optimizers that suit any given problem is presented. [19]. The authors propose the usage of Dynamic Structured Grammatical Evolution [20] for automatically creating SGD variations for each specific problem. To that end, 8 different operators (e.g., addition and subtraction) and multiple parameters can be combined into a single expression which determines the manner in which the gradients computed by the backpropagation algorithm will affect the weights in the DNN. Two variations of the proposal are tested against manually created SGD variations (Adam, SGD with Nesterov momentum, and RMSProp) in the task of training a DNN architecture (which was itself previously evolved) for two problems, MNIST and CIFAR10 [21].

The work in [4] proposes another GAN evolutionary approach, one based on a genetic algorithm. Despite most of the mutation operators being directed towards the architecture of the model, other critical aspects of the model, such as the loss function and the random noise fed to the generator, are also included in the evolutionary procedure. Within the NE framework, the capacities of the GANs are gauged by measuring the inverted generational distance (IGD) [22] between a set of points known to be in the Pareto set of a bi-objective problem and the points generated by the GAN, after being trained to generate points from that set. As a reference for the ground truth, the benchmark described in [23] was used, which provides difficult optimization problems along with their corresponding solutions. After the GANs were evolved specifically for those problems, the transferability of the evolved structures was validated when they were successfully applied to the 2D 8-Gaussian approximation problem [14].

[1] Available in https://github.com/tensorflow/models/blob/master/research/domain_adaptation/domain_separation/losses.py

In this work, we use the GAN structures evolved in the work presented above, whose transferability to the 2D 8-Gaussian approximation problem has already been tested.

Recent work [24]–[27] on DNNs has increasingly emphasized the importance of model transferability across problem domains as an efficient way to reduce the time-consuming problem of network design.

III. Role of the optimizer in the GAN learning procedure

In the analysis carried out in [4], several components of the GAN structure were included in the NE runs, from the architecture of the DNNs, to the loss function used to train the model. The SGD algorithm, a component which can be as important as the rest of component of a GAN, however, was not taken into account. In this paper, we undertake the task of testing to what extent this aspect of the DNN training is relevant to the final product. Furthermore, as it has been introduced in Section II-A, the two DNNs composing a GAN are usually trained using different loss functions, which opens the door to an additional question: Can a GAN be successfully trained using two different optimizers?

In [4], the authors evolved GANs for a total of 16 problems, consisting of eight functions and two different number of variables. Additionally, the experiments consisted of 30 separate runs of the same algorithm. One of these problem sizes contained 784 variables (inspired in the number of variables in the MNIST problem). We select the best GAN (in terms of IGD) from each run from the problems with 784 variables, and retrain the GANs so that they are able to produce entire distributions in each sample. Because the GANs are prepared to generate 784 values each time, we can obtain $784/2 = 392$ total points to represent a distribution.

IV. Experiments

A. Experimental framework

In this paper, we consider four different methods: RMSProp [28], FTRL [29], Adagrad [30], and gradient descent with momentum [31]. Additionally, combinations of FTRL and RMSProp, two optimizers which provided good quality and interesting results respectively, are combined into single GANs, by training both networks with different combinations of the algorithms.

The methodology to assess the quality of the GANs when trained as explained in Section III using these different SGD algorithms is based on 16 different samples obtained from the GANs after the training phase. Because each experiment was repeated 30 times, for each function available in the experimentation $30 \times 16 = 480$ total sets of points are obtained.

B. Preliminary visualization

Some examples of the results obtained by the GANs can be visualized in Figures 2 and 3. The first one displays random generations of one GAN extracted from an evolutionary run carried out with one of the eight functions, F1, (subfigures in the left-hand side) and another one from another function run (in the right-hand side), F2. Each row of subfigures corresponds with one optimization method, single optimizers in Figures 2 and combined ones in 3. The approach of the different optimizers to the same GAN structure varies vastly, according to this preliminary visualization.

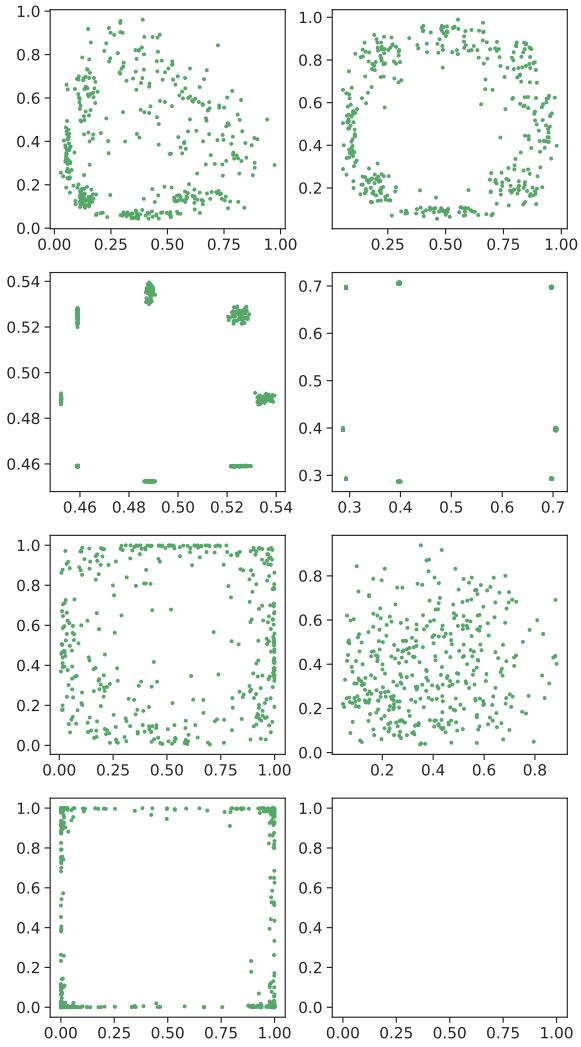


Fig. 2: Results of the four single optimizers for two two structures. From top to bottom: RMS, FTRL, Adagrad, Momentum. The subfigures in the left-hand side correspond with GANs extracted from a evolutionary run carried out for F1, whereas the ones in the right-hand side correspond with F2.

Regarding Figure 2, the GANs trained with the RMSProp algorithm (two subfigures on the top) appears to focus on different areas each time. While the one obtained from a F1 run focuses on accurately representing the distribution with low values in both x and y axis first, the other one, obtained from a F2 run approaches the challenge by firstly avoiding generating point in the center and corners of the grid. Both can be valid approaches, as in both cases the final result can be satisfactory.

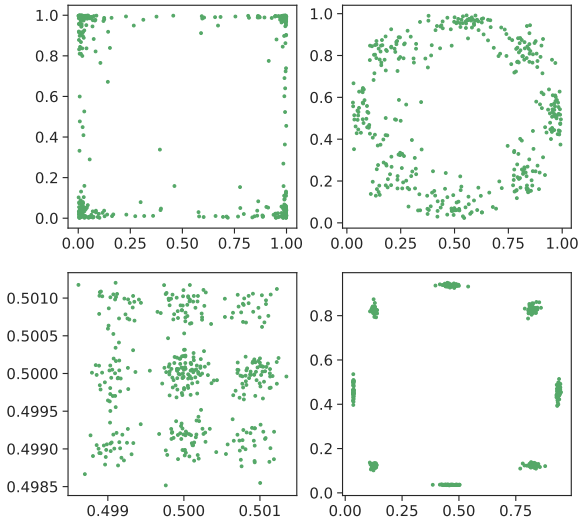


Fig. 3: Results of two combined optimizers for two structures. Top: RMS-FTRL. Bottom: FTRL-RMS.

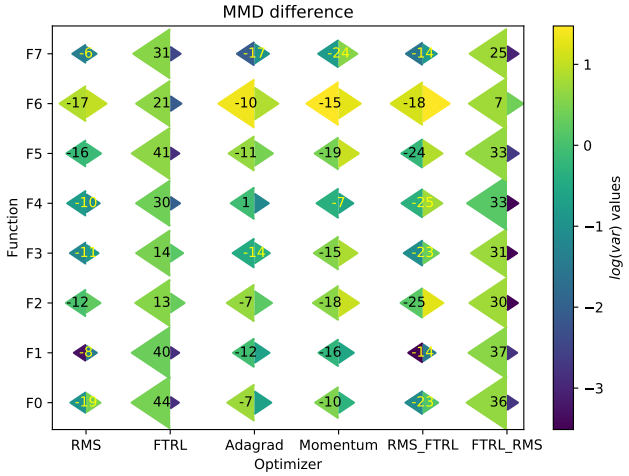


Fig. 4: Mean MMDs computed, with the variances annotated.

GANs trained with FTRL, however, show a totally different behavior. Apparently, instead of reducing the probability of producing points from areas with non-existent density in the original data set, they immediately capture the *structure* of the problem, as they right away are able to produce eight clusters of points. However, their generations are not totally accurate, as, instead of generating points over all the space [0-1], it is limited to [0.45-0.55], in the case of the GAN obtained from the F1 run. The case of the GAN from the F2 run is even more interesting. Although the scale is closer to the [0-1] shown in the original data (limited to [0.29-0.71]) the eight clusters of points do not represent the same *shape* shown in the training data. They also respect proportionality and symmetry, using [0.5,0.5] as the reference center point.

The other two optimizers, AdaGrad and Momentum (the four subfigures in the lower part of Figure 2), were unable to produce good quality results.

Regarding the combined optimization methods, those shown in Figure 3, results equally disparate and interesting can be observed. The GAN whose discriminator was trained with the RMSProp and generator with FTRL (figures in the top) show different behaviors already observed with the individual optimizers, the GAN from the F1 run was unable to reproduce the distribution, whereas the one from F2 appears to approach the problem by slowly concentrating the generated points on the same areas of high density in the training data.

The GANs trained with the opposite combination, however, show different results. The GAN obtained from the F2 run was able to *correctly* produce points in terms of the number of modes, and the location of the modes regarding the low values. The location of the points with respect to the higher values, however, is not totally accurate, as they were upper limited to ~ 0.88 in both scales, identifying a new pattern. The strangest observed behavior, however, is the one shown by the GAN obtained from the F1 run, as it generates data from nine modes, in a minuscule scale, [0.498, 0.5012].

From this preliminary observation, we can deduce that RMSProp is a reliable GAN optimizer, as in most cases in which it was involved, GANs were able to reproduce similar distributions to the target. Additionally, we identify FTRL as a optimizer with an anomalous behavior, as it produces results in different scales from that of the original data, and it was able to produce a nine-mode distribution, instead of eight.

As can be seen in the last figure in the bottom left, sometimes, GANs were unable to be correctly trained, and ended producing invalid generations. The Momentum optimizer produced most of these cases ($\sim 20\%$ of the training procedures), although RMSProp also had some ($\sim 8\%$).

C. Main analysis

We now perform a more rigorous analysis regarding the number of GANs. The 480 sets of samples are from the GANs corresponding to each function are individually compared to a target set of points, which is drawn from the same distribution from which the set used to train the GANs was obtained. The maximum mean discrepancy (MMD) [15] is the metric of choosing for this comparison task.

Next, the quality of the GANs trained with the different optimization algorithms is assessed using the MMD (lower values represent better approximation), which has already been used in this problem [5]. Malfunctioning GANs are penalized with the largest valid computed MMD.

The analysis performed consists of obtaining the mean and the variance of the 480 different metric values obtained (per function and optimization method), and these can be visualized in Figure 4. In the figure, results discriminated by optimizers (x axis) and functions used to evolve the GANs (y axis) are shown. Per each combination, two triangles are displayed, which size represents the mean of the 480 MMD measurements. The one in the left corresponds with a pure comparison between the generated points and the target set.

Function	RMS		FTRL		AdaGrad		Momentum		RMS_FTRL		FTRL_RMS		Averages	
	Reg.	Norm.	Reg.	Norm.	Reg.	Norm.	Reg.	Norm.	Reg.	Norm.	Reg.	Norm.	Reg.	Norm.
F1	81	111	-112	117	-46	-23	16	20	25	3	-208	16	-40.67	40.67
F2	86	114	-121	113	-4	-8	-1	-6	17	-4	-199	13	-37	37
F3	69	141	-89	55	-29	6	-14	-3	21	1	-174	16	-36	36
F4	74	125	-110	89	-7	21	-18	-11	8	-1	-185	15	-39.67	39.67
F5	62	114	-74	137	-7	26	15	26	-27	-23	-238	-11	-44.83	44.83
F7	66	84	-97	135	-8	4	6	10	8	-1	-204	-3	-38.17	38.17
F8	47	91	-11	155	-77	-1	-31	29	-67	-45	-112	22	-41.83	41.83
F9	93	185	-116	80	-18	-7	-23	-17	39	16	-213	-19	-39.67	39.67
Average	72.25	120.63	-91.25	110.13	-24.5	2.25	-6.25	6	3	-6.75	-191.63	6.125	-39.72	39.72

TABLE I: Number of statistical comparisons (made between same GAN structures) won by each optimization method with and without normalized results, per function.

In the previous section we observed that, in some cases, GANs could produce high quality results, but in the incorrect scale. As a more interesting analysis, we manually scaled all generations (some of which did not need to) to [0-1], and recomputed the two metrics. The rationale behind this is evaluating the capacity of the models to generate the relative distribution of the clusters, even if the distribution lies in a distorted scale. The mean of this second MMD computation is represented in the size of the triangle of the right. The color of the triangles represents the variance found in the sets of 480 measurements. Additionally, a ranking of mean metric values is computed, for each version of the generations (with and without normalization). The ranking position of each GAN/optimizer combination with normalized samples is subtracted from that of combinations without normalization. The resulting number is displayed inside the triangles. The larger this number, the more positions a combination escalated in the ranking by normalizing its generations.

As can be seen in the figure, and could have been expected, the MMDs from the normalized generations are, in general, lower than those computed from the samples without being normalized. It can be clearly seen that, without considering normalization, the GANs which used the FTRL for training the discriminator offered a considerably worse performance. In that case, the RMSProp optimizer appears to be the safest choice, as it offered low values in both mean and MMD variance. When analyzing normalized results, however, the perspective changes altogether. In this case, it is the FTRL-trained discriminators which compose the top performing GANs. The ascension in the ranking observed in the FTRL-trained GANs backs this observation, as these models performed the higher improvement, causing the rest of optimizers to drop positions, accordingly.

As a complement to these findings, we also offer some statistical backing to the results. We firstly select all the MMD values obtained from each GAN structure, i.e., $2\ normalization\ approaches \times 8\ optimizers = 16$ sets of 16 measurements (these corresponding with the amount of times a GAN is sampled). After the Wilcoxon signed-rank test rejected the null hypothesis of differences not existing between the related sets of values (p-value=0), we employ the Dunn post-hoc test to perform pairwise comparisons between all the samples generated from the same GAN structure, including comparisons between sets of MMD values obtained with and without normalizing samples. Each time the Dunn test indicated significant differences between MMD values ($pvalue \leq 0.05$), the means of the 16 MMD values were compared, and the combination with the lower MMD average scored one point, whereas the one with the higher one got a point subtracted. Therefore, higher scores denote better performances. After all comparisons, all points are added together taking into account the functions, optimizers, and whether normalization was employed in the samples. These results are displayed in Table I, which also contains a final row with the average points for each method, and columns for each function and normalization approach.

As can be seen in the columns showing averages per function and normalization approach (last two), performing normalization of the obtained samples offered significantly improved results, as they scored ~ 40 more positive points on average, according to the statistical test. Additionally, in this case, it seems that the GANs evolved with F5 were the ones with the highest level of transferability. Focusing on the final row of the table, it can be seen that the RMSProp optimizer offered the best results among the non-normalized approaches, and maintained that lead in the sample-normalizing variant, albeit narrowly. The FTRL optimizer went from second-to-last without normalization, to almost obtaining the best results of all optimizers. Something similar happened to the FTRL_RMSProp combination, which went from last to third, although with a large margin to the second position, and very close to the rest of approaches. This highlights that not only the optimizer used to train the GAN model is important, but the optimizer for each DNN within the model is.

V. Conclusions

One of the open questions in the analysis of the effectiveness of neuro-evolutionary algorithms is the interplay between the structural search as conducted by neural architecture search methods and the model optimization as conducted with gradient optimizers. This question is particularly difficult to address for multi-network models focused on data generation, since data generation metrics can be tricky, and multi-network models can use different gradient optimizers for each component. In this paper we have presented a first step into evaluating this relationship using the GAN as an exemplar model.

We have shown that, even for evolved structures which are more efficient than random network structures [4], the choice of the gradient optimizer can represent important gains in the quality of the generated data. The gains can be either in terms of similarity to the absolute positions of points from the original data, or in terms of the relative position of the clusters of points in the generated data to the original relative positions of the clusters. Interestingly, we have observed that while most of the gradient optimizers are precise for absolute data generation, others, in particular the FTRL optimizer, are more appropriate for reproducing the relative position of the clusters. Furthermore, we have shown evidence that incorporating the choice of the gradient optimizer has a different effect on the discriminator and generator components of the GANs and that it can lead to important gains in neural architecture searches.

VI. Acknowledgements

This work has received support from the TIN2016-78365-R (Spanish Ministry of Economy, Industry and Competitiveness), PID2019-104966GB-I00 (Spanish Ministry of Science and Innovation), IT-1244-19 (Basque Government) and KK-2020/00049 (part of the project 3KIA, funded by the SPRI-Basque Government through Elkartek) programs. We acknowledge the support of NVIDIA Corporation with the donation of a Titan X Pascal GPU.

References

[1] I. Loshchilov and F. Hutter, "CMA-ES for hyperparameter optimization of deep neural networks," *arXiv preprint arXiv:1604.07269*, 2016. [Online]. Available: https://arxiv.org/abs/1604.07269

[2] R. Miikkulainen, J. Liang, E. Meyerson, A. Rawal, D. Fink, O. Francon, B. Raju, H. Shahrzad, A. Navruzyan, N. Duffy, and others, "Evolving deep neural networks," in *Artificial Intelligence in the Age of Neural Networks and Brain Computing*. Elsevier, 2019, pp. 293–312. [Online]. Available: https://doi.org/10.1016/B978-0-12-815480-9.00015-3

[3] M. Suganuma, S. Shirakawa, and T. Nagao, "A genetic programming approach to designing convolutional neural network architectures," in *Proceedings of the Genetic and Evolutionary Computation Conference*. ACM, 2017, pp. 497–504.

[4] U. Garciarena, A. Mendiburu, and R. Santana, "Analysis of the transferability and robustness of GANs evolved for Pareto set approximations," *Neural Networks*, vol. 132, pp. 281–296, 2020, publisher: Elsevier.

[5] C. Wang, C. Xu, X. Yao, and D. Tao, "Evolutionary Generative Adversarial Networks," *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 6, pp. 921–934, 2019.

[6] V. Costa, N. Lourenço, and P. Machado, "Coevolution of Generative Adversarial Networks," in *International Conference on the Applications of Evolutionary Computation (Part of EvoStar)*. Springer, 2019, pp. 473–487.

[7] J. Toutouh, E. Hemberg, and U.-M. O'Reilly, "Spatial Evolutionary Generative Adversarial Networks," in *Proceedings of the Genetic and Evolutionary Computation Conference*. New York, NY, USA: Association for Computing Machinery, 2019, pp. 472–480, event-place: Prague, Czech Republic.

[8] U. Garciarena, N. Lourenço, P. Machado, R. Santana, and A. Mendiburu, "On the Exploitation of Neuroevolutionary Information: Analyzing the Past for a More Efficient Future," *arXiv e-prints*, pp. arXiv–2105.12 836, 2021.

[9] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative Adversarial Nets," in *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*. Cambridge, MA, USA: MIT Press, 2014, pp. 2672–2680, event-place: Montreal, Canada.

[10] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014. [Online]. Available: https://arxiv.org/abs/1412.6980

[11] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, "On the importance of initialization and momentum in deep learning," in *International conference on machine learning*. PMLR, 2013, pp. 1139–1147.

[12] M. D. Zeiler, "Adadelta: an adaptive learning rate method," *arXiv preprint arXiv:1212.5701*, 2012.

[13] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training GANs," in *Advances in Neural Information Processing Systems*, 2016, pp. 2234–2242.

[14] L. Metz, B. Poole, D. Pfau, and J. Sohl-Dickstein, "Unrolled generative adversarial networks," *arXiv preprint arXiv:1611.02163*, 2016.

[15] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, "A Kernel Two-Sample Test," *Journal of Machine Learning Research*, vol. 13, no. 25, pp. 723–773, 2012. [Online]. Available: http://jmlr.org/papers/v13/gretton12a.html

[16] A. Al-Dujaili, T. Schmiedlechner, E. Hemberg, and U.-M. O'Reilly, "Towards distributed coevolutionary GANs," in *AAAI 2018 Fall Symposium*, 2018.

[17] M. Mitchell, "Coevolutionary learning with spatially distributed populations," in *Computational intelligence: principles and practice*, D. B. Fogel, Ed. IEEE Computational Intelligence Society, 2006. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.224.6254&rep=rep1&type=pdf

[18] Y. LeCun, C. Cortes, and C. Burges, "MNIST handwritten digit database," *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, vol. 2, 2010.

[19] P. Carvalho, N. Lourenço, and P. Machado, "Evolving Learning Rate Optimizers for Deep Neural Networks," *arXiv preprint arXiv:2103.12623*, 2021.

[20] N. Lourenço, F. B. Pereira, and E. Costa, "Unveiling the properties of structured grammatical evolution," *Genetic Programming and Evolvable Machines*, vol. 17, no. 3, pp. 251–289, 2016, publisher: Springer.

[21] A. Krizhevsky, "Learning multiple layers of features from tiny images," Master's thesis, Department of Computer Science, University of Toronto, 2009. [Online]. Available: https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf

[22] C. A. Coello Coello and M. Reyes Sierra, "A Study of the Parallelization of a Coevolutionary Multi-objective Evolutionary Algorithm," in *MICAI 2004: Advances in Artificial Intelligence*, R. Monroy, G. Arroyo-Figueroa, L. E. Sucar, and H. Sossa, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 688–697.

[23] H. Li and Q. Zhang, "Multiobjective Optimization Problems with Complicated Pareto Sets, MOEA/D and NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 2, pp. 284–302, 2008.

[24] Y. Cui, Y. Song, C. Sun, A. Howard, and S. Belongie, "Large scale fine-grained categorization and domain-specific transfer learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4109–4118.

[25] S. Kornblith, J. Shlens, and Q. V. Le, "Do better imagenet models transfer better?" in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2661–2671.

[26] A. S. Morcos, H. Yu, M. Paganini, and Y. Tian, "One ticket to win them all: generalizing lottery ticket initializations across datasets and optimizers," *arXiv preprint arXiv:1906.02773*, 2019. [Online]. Available: https://arxiv.org/abs/1906.02773

[27] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le, "Learning transferable architectures for scalable image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8697–8710.

[28] T. Tieleman and G. Hinton, *Lecture 6.5—RmsProp: Divide the gradient by a running average of its recent magnitude*, 2012, published: COURSERA: Neural Networks for Machine Learning.

[29] H. B. McMahan, G. Holt, D. Sculley, M. Young, D. Ebner, J. Grady, L. Nie, T. Phillips, E. Davydov, D. Golovin, and others, "Ad click prediction: a view from the trenches," in *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2013, pp. 1222–1230.

[30] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization." *Journal of Machine Learning Research*, vol. 12, no. 7, 2011.

[31] N. Qian, "On the momentum term in gradient descent learning algorithms," *Neural networks*, vol. 12, no. 1, pp. 145–151, 1999, publisher: Elsevier.

Aplicación de técnicas de aprendizaje profundo al reconocimiento óptico de partituras SATB

Martin Morita-Hernández*, Francisco Fernandez de Vega†
Departamento de Tecnología de los Computadores
y de las Comunicaciones
Universidad de Extremadura
Mérida, España
Email: *martin.morita.13@gmail.com †fcofdez@unex.es

Juan Villegas Cortez
Departamento de Sistemas
Universidad Autónoma Metropolitana, Unidad Azcapotzalco
Cd. de México, México
Email: juanvc@azc.uam.mx

***Resumen*—El reconocimiento óptico de partituras es un problema que forma parte del ámbito OMR (Optical Music Recognition). La gran variedad de elementos presentes en una partitura, que incluye clave, compás, tempo, dinámica, articulación, repeticiones, además de las notas, duración y alteraciones, hacen de su reconocimiento óptico un problema interesante en el dominio de la inteligencia artificial. Este artículo aborda este difícil problema utilizando técnicas de aprendizaje profundo y contextualizando en el aprendizaje de estudiantes de armonía, que realizan sus ejercicios de forma manuscrita sobre papel. Los resultados preliminares obtenidos, con unos porcentajes de acierto de alrededor del 95 % en la clasificación correcta de los elementos de la partitura en la fase de entrenamiento, nos hacen vislumbrar la posibilidad de crear una herramienta útil tanto para profesores como para alumnos de armonía.**

***Index Terms*—Aprendizaje profundo, OMR, Reconocimiento óptico.**

I. Introducción: OMR

El reconocimiento óptico de información musical (OMR) es un área que goza de bastante interés entre las aplicaciones de la inteligencia artificial a la música.

La Sociedad Internacional para la Recuperación de Información Musical (ISMIR, International Society for Music Information Retrieval) lo considera como una de las áreas relevantes, y difíciles de abordar. Frente al reconocimiento óptico de caracteres, en el que cada carácter tiene un valor independientemente de su posición dentro de un documento, (e.g., la letra "a" siempre representa la misma información en un texto), las notas musicales poseen un valor que depende no solo de la grafía de la misma, sino de su posición sobre el pentagrama. Pero además de esto, una partitura posee mucha información adicional que se refieren al modo en que la nota musical debe emitirse con el instrumento, incluyendo dinámica (*fuerte, piano,...*), articulación (*ligado, stacatto,...*), repeticiones (*saltos, codas, da capo...*), claves (*Sol, Fa, Do...*), compás (2 *por* 4, 4 *por* 4, 6 *por* 8, ...), etc.

Así, y aunque en principio las técnicas disponibles para el reconocimiento óptico de caracteres puedan ser aplicables en el nuevo marco, la realidad es que se necesitan técnicas adicionales para poder obtener toda la información relevante, si el objetivo es convertir una partitura en papel a un formato de partitura estándar, tal como MIDI[1] o MusicXML[2].

Este artículo explora técnicas disponibles para realizar el reconocimiento de partituras manuscritas en el contexto de los ejercicios de armonía a cuatro voces, cuyas reglas son aprendidas por los estudiantes de tercer y cuarto curso de los conservatorios profesionales de música en España. Tal como describimos más adelante, el objetivo es desarrollar un módulo para que la aplicación y herramienta Sharpmony [3] permita a profesores y estudiantes capturar la información de cualquier ejercicio de armonía escrito a mano sobre una libreta pautada.

En la sección II describimos el tipo de ejercicio manuscrito al que nos enfrentamos, y revisamos las propuestas disponibles en OMR que puedan servir como punto de partida al trabajo que hemos desarrollado. En la sección 3 presentamos en detalle la metodología aplicada, y su implementación con los resultados los discutimos en la sección 8. La sección V describe las mejoras previstas y su aplicación a la herramienta *Sharpmony*. Finalmente en la sección VI compartimos las conclusiones y el trabajo futuro.

II. Armonía SATB

En música clásica, el estudio de armonía es uno de los elementos claves para que los alumnos desarrollen sus habilidades en composición. La música occidental ha desarrollado y evolucionado una serie de reglas armónicas que son trabajadas por todos los estudiantes matriculados en los conservatorios profesionales de música que estudian la música occidental. Estas reglas, que comenzaron a desarrollarse más o menos

[1]Acrónimo de Musical Instrument Digital Interface(Interfaz digital para instrumentos musicales). Protocolo para la transmisión de datos musicales entre componentes digitales, como los sintetizadores y las tarjetas de sonido de las computadoras. Britannica Escolar, s.v. "MIDI", consultado el 6 de mayo, 2021, https://bidi.uam.mx:8429/levels/academica/article/MIDI/421779.

[2]MusicXML es un formato abierto de notación musical basado en XML. Consultado el 6 de mayo, 2021, https://www.musicxml.com/

[3]*Sharpmony* es la primera herramienta basada en Inteligencia Artificial para ayudar a los estudiantes de Armonía y sus profesores. Utilizando las últimas tecnologías de IA desarrolladas en la Universidad de Extremadura, integra una aplicación móvil y un portal web que permitirá al estudiante revisar ejercicios de armonía clásica (SATB), y realizar armonización automática de un tiple o bajo cifrado. Consultado el 6 de mayo, 2021, https://sharpmony.unex.es

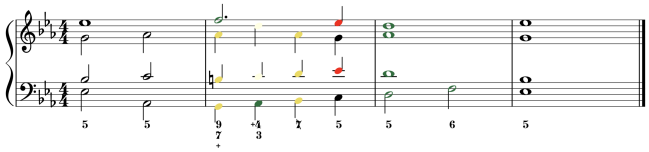


Figura 1. Ejercicio corregido.

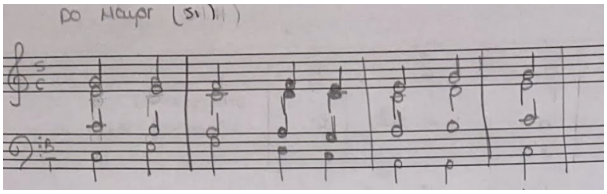


Figura 2. Ejercicio manuscrito de armonía a cuatro voces.

formalmente en el renacimiento, se establecen de forma definitiva en el Barroco, con el estudio del contrapunto y la fuga, siendo Bach la figura de referencia del periodo. Además de música instrumental, Bach desarrolla toda una serie de composiciones corales, en las que la superposición de voces requiere de habilidad y maestría en el desarrollo armónico.

Así, en el periodo que le sigue, el clasicismo, se afianza el estudio de la armonía coral a cuatro voces, integrada por cuatro voces: Soprano, Contralto, Tenor y Bajo (SATB); y surgen nuevas reglas que marcan la novedad de la música del periodo frente a lo que se considera antiguo, por pertenecer a estilos pasados, como el Barroco o Renacimiento.

Es de este periodo, el clasicismo, del que se toma el conjunto de reglas que actualmente se estudia en las asignaturas Armonía 1 y 2, aplicadas a corales SATB en los curso tercero y cuarto de las enseñanzas profesionales de música, y que continúan perfeccionando posteriormente los estudiantes con el estudio del contrapunto y otras técnicas de composición más modernas en cursos posteriores del grado profesional, así como en el grado superior de música.

El método de estudio consiste por tanto en tomar como punto de partida una melodía que se asigna a la Soprano, o un bajo cifrado, y el estudiante debe componer el resto de las voces siguiendo unas reglas que indican lo que está permitido hacer y lo que no. Las reglas afectan tanto a los intervalos entre voces, como a los movimientos melódicos que cada voz desarrolla en combinación con el resto de voces.

La Figura 1 muestra un ejemplo de ejercicio de armonía coral desarrollado por un estudiante, con algunas notas marcadas con colores, correspondiente a notas que no cumplen las reglas de armonía clásica. Concretamente aparecen marcados errores correspondientes a *Distancias mayor que octava entre voces (amarillo claro) Cifrado Incorrecto (amarillo mostaza), Quintas-Octavas paralelas (rojo) y acordes incorrectos (verde oliva)*.

II-A. Inteligencia artificial y armonía SATB

Son varios los problemas asociados a la armonía SATB en los que pueden aplicarse diferentes técnicas de Inteligencia Artificial. Podemos destacar:

- Corrección automática de ejercicios SATB.
- Composición automática de corales SATB.
- Reconocimiento automático de ejercicios SATB escritos a mano (Optical Music Recognition, OMR).

Como ejemplo del primer punto, lo que mostramos en la figura 1 corresponde a una corrección realizada de forma automática con la herramienta *Sharpmony*, herramienta que incorporando Inteligencia Artificial está ya siendo usada por varias instituciones para ayuda a la docencia de armonía, y que esperamos pueda incorporar en breve las metodologías que presentamos a continuación.

En este trabajo nos vamos a ocupar del tercero de los puntos detallados anteriormente, y que corresponde con el reconocimiento automático de partituras con ejercicios SATB. En la figura 2 mostramos un ejercicio típico desarrollado por un estudiante en su libreta pautada. Como puede observarse el ejercicio contiene indicaciones del compás, la tonalidad (en este caso al ser Do Mayor el ejercicio no contiene armadura) y las diferentes figuras de cada compás y voz, incluyendo redondas, blancas y negras.

II-B. Reconocimiento automático de partituras

El problema OMR es bien conocido en la comunidad de Recuperación Automática de Información Musical (Music Information Retrieval, MIR). Aunque existen dos versiones del mismo, el que corresponde a partituras impresas y aquel dedicado a partituras manuscritas, el más interesante y a la vez difícil es el segundo caso, que corresponde con lo que mostramos anteriormente en la figura 2.

Las recientes investigaciones sobre redes neuronales convolucionales (CNN, Convolutional Neuronal Networks) por medio del aprendizaje profundo (Deep Learning), han dado pie a que se propongan varios enfoques OMR. Durante este período, han surgido diversos algoritmos de detección de objetos (Object Detection) que compiten entre sí, como YOLO [1], SSD [2] y RetinaNet [3], siendo clasificados como modelos de detección de una etapa, por lo que tienen una velocidad de detección más rápida. También existen modelos de detección de dos etapas, como Faster R-CNN [4], R-CNN [5], y R-FCN [6], tienen niveles de precisión de detección más altos pero velocidades más lentas. Mientras que algunos optimizan la precisión, otros se esfuerzan por lograr un alto rendimiento.

Se ha demostrado que los procedimientos tradicionales de segmentación y clasificación de símbolos funciona bien en partituras musicales impresas [7]. Pero cuando se consideran imágenes de partituras escritas a mano, estos sistemas tienden a fallar, porque los errores se propagan a lo largo del entrenamiento, por ejemplo, un error de segmentación podría causar objetos detectados incorrectamente. Algunos investigadores han hecho intentos para superar esta limitación haciendo uso de CNN, por ejemplo, en [8] Pacha y sus colegas para detectar notas utilizan tres conjuntos de datos distintos

DeepScores [9], MUSCIMA++ [12] y Capitan collection [10], además de sugerir una línea base para la detección general de objetos dentro de la partitura, haciendo uso de tres modelos distintos de detección de objetos Faster R-CNN, U-Net [11] y RetinaNet. Los resultados de las pruebas de cada modelo se dieron en términos de precisión promedio.

Además Hajič y sus colegas proponen la detección directa de objetos musicales con CNN [13], quienes sugieren una adaptación de Faster R-CNN con un mecanismo de propuesta de región personalizada basado en el esqueleto morfológico para detectar las cabezas de las notas, no requiere la eliminación del pentagrama y es aplicable a una variedad de estilos de escritura y niveles de complejidad musical. En investigaciones posteriores sugieren identificar las notas en dos etapas, haciendo uso de la arquitectura U-Net [14]. Durante la primera etapa, la imagen de la puntuación de entrada es segmentada como imagen binaria utilizando el modelo de segmentación semántica, y el problema de detección general se descompuso en un conjunto de problemas de clasificación de píxeles binarios, y luego se utilizó el detector de componentes conectados para derivar la propuesta de detección final. El experimento tuvo como objetivo el conjunto de datos MUSCIMA++, que muestra los resultados de detección de símbolos en términos de puntuaciones.

Tuggener y sus colegas hicieron uso de ResNet [15] para predecir mapas de energía con el fin de obtener la ubicación, la clase y el cuadro delimitador de cada símbolo contenido dentro de una partitura, sin necesidad de recortar por secciones las imágenes de pentagramas [16]. Este método es efectivo detectando símbolos pequeños, pero genera cuadros delimitadores inexactos y clases no contenidas en el sistema.

Observamos algunos inconvenientes en los métodos anteriores. Después de los enfoques basados en la detección de objetos, se requieren pasos adicionales para detectar los símbolos y obtener el tono de la nota.

En este artículo, proponemos el uso de un nuevo clasificador de redes neuronales convolucionales (CNN) denominado Mask R-CNN [17], que tiene el potencial de lograr una mejor precisión de reconocimiento en las partituras escritas a mano, simplificando el proceso y prediciendo una máscara binaria para cada clase de forma independiente.

III. Metodología

A continuación presentamos la metodología propuesta para reconocimiento de partituras escritas a mano, tal como se aprecia en la Figura 3, esta consta de cuatro etapas generales.

En primer lugar hemos procedido a crear nuestro repositorio formado por fotografías de las partituras escritas a mano, a las que se aplica el proceso de segmentación detallado más abajo.

A continuación estas partituras ya segmentadas han sido utilizadas para entrenar una red neuronal convolucional profunda, pre-entrenada, y finalmente se obtiene una clasificación de los símbolos varios contenidos en cada imagen del repositorio. Este procedimiento se realiza en una primera etapa de entrenamiento para nuestro problema, posteriormente se realiza la etapa de prueba sobre imágenes desconocidas para la red.

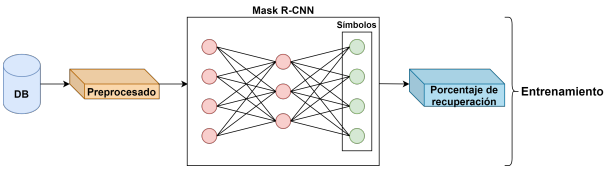


Figura 3. Metodología.

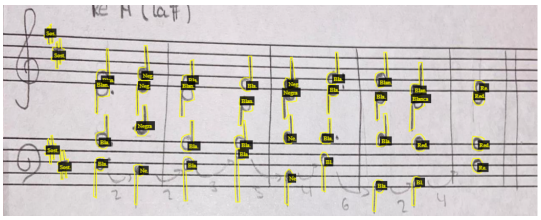


Figura 4. Ejemplo de preprocesamiento del conjunto de datos, para cada partitura SATB.

En las siguientes líneas damos detalles de las etapas de la metodología para facilitar la comprensión de nuestro trabajo.

III-A. Conjunto de datos (DB)

Para entrenar el detector de notas musicales, se elaboró nuestro propio conjunto de datos utilizando algunas técnicas de *data augmentation*, como los son recortar y escalar, aplicadas a imágenes de partituras SATB suministrada por los profesores de armonía que colaboran en el proyecto, obteniendo 100 imágenes que contienen más de 3000 anotaciones a nivel de símbolo, que incluyen redondas, blancas y negras, así como la armadura de la partitura, que sirve para detección de la tonalidad, indicaciones del compás, y las diferentes figuras de cada compás y voz, realizadas por estudiantes de conservatorios en sus libretas pautadas.

III-B. Preprocesamiento

A partir del conjunto de datos creado con las imágenes de partituras obtenidas de los alumnos, fueron adaptadas de modo que todas tuvieran el mismo tamaño y formato. El proceso se dividió en dos partes. Primero, para entrenar eficientemente el detector de objetos la implementación evaluada coge imágenes en color como entrada, las cuales deben tener el mismo tamaño, por tal motivo han sido redimensionadas en 1280×720 píxeles. Segundo, se utilizó VGG Image Annotator (VIA), para identificar dentro de la imagen de la partitura las ubicaciones de cada nota, así como el símbolo al que pertenecen. Con toda la información generada, disponemos de imágenes de partituras acompañadas por un archivo en formato JSON, que contiene la ubicación de cada una de las notas dentro de la imagen y sus respectivas etiquetas, que se representan con un polígono delimitador adaptado a la forma de cada símbolo, como se muestran en la Figura 4.

III-C. Aplicación de la CNN

Para el experimento, se evaluó la arquitectura Mask R-CNN, presentada en 2017, es una extensión de Faster R-CNN

que añade una rama para predecir máscaras segmentadas en cada región de interés, siendo esta paralela a las tareas de identificación y localización. Esto es, si Faster R-CNN tenía como salida las clases y los cuadros delimitados, Mask R-CNN añade a estos una máscara binaria haciendo uso de una operación *RoI Align* (RoI, Region of Interest) para cada región de interés.

Mask R-CNN se divide en tres etapas, tal como se muestra en la Figura 5. Primero, una red se encarga de generar los mapas de características de las imágenes de entrada. En segundo lugar, los mapas de características generados son enviados a la red de propuesta de región (RPN) para generar regiones de interés (RoI). En tercer lugar, las RoI generadas por RPN se mapean para extraer las características correspondientes para la clasificación de objetos y la segmentación de instancias. Este proceso genera las puntuaciones de clasificación, los cuadros delimitadores y las máscaras de segmentación.

El objetivo de nuestro modelo no es detectar todos los símbolos en la partitura, sino proponer un nuevo método de reconocimiento de objetos musicales, centrándose en el reconocimiento de notas, utilizando los símbolos más comunes dentro de una partitura SATB. Aquí, no necesitamos detectar simbología compleja. Los símbolos representativos se muestran en la Figura 6.

III-D. Porcentaje de recuperación

La pérdida de entrenamiento de Mask R-CNN consta de tres partes principales: la pérdida de clasificación (L_{cls}) ecuación 1, localización (L_{box}) ecuaciones 2, 3, y máscara de segmentación (L_{mask}) ecuación 4. La pérdida total de entrenamiento se puede calcular mediante la ecuación 5.

$$L_{cls}(p_i^*\ ,\ p_i) = -\ log(p_i^*\ p_i) \tag{1}$$

$$L_{box}(t_i\ ,\ t_i^*) = L_1^{smooth}(t_i^* - t_i) \tag{2}$$

$$L_1^{smooth}\ (x)\ =\ \begin{cases} 0{,}5x^2 & if\ \ |x| < 1 \\ |x| - 0{,}5 & en\quad otro\quad caso \end{cases} \tag{3}$$

$$L_{mask}(s_i\ ,\ s_i^*) = -(s_i^* \log s_i + (1 - s_i^*) \log(1 - s_i)) \tag{4}$$

$$L(p_i,\ p_i^*,\ t_i,\ t_i^*,\ s_i,\ s_i^*)\ =\ \frac{1}{N_{cls}} \sum_i L_{cls}\ (p_i\ ,\ p_i^*) + \frac{\lambda}{N_{box}} \sum_i\ p_i^*\ L_{box}\ (t_i\ ,\ t_i^*)\ +\ \frac{\gamma}{N_{mask}} \sum_i L_{mask}\ (s_i\ ,\ s_i^*) \tag{5}$$

En la ecuación 5, N_* representa el número de cuadros delimitadores correspondientes. Los parámetros λ y γ equilibran las pérdidas de entrenamiento de la regresión y la rama de máscara.

Además, p_i representa la probabilidad predicha de que el cuadro delimitador i sea un objeto, y p_i^* representa la probabilidad de verdad básica (binaria) de si el cuadro delimitador

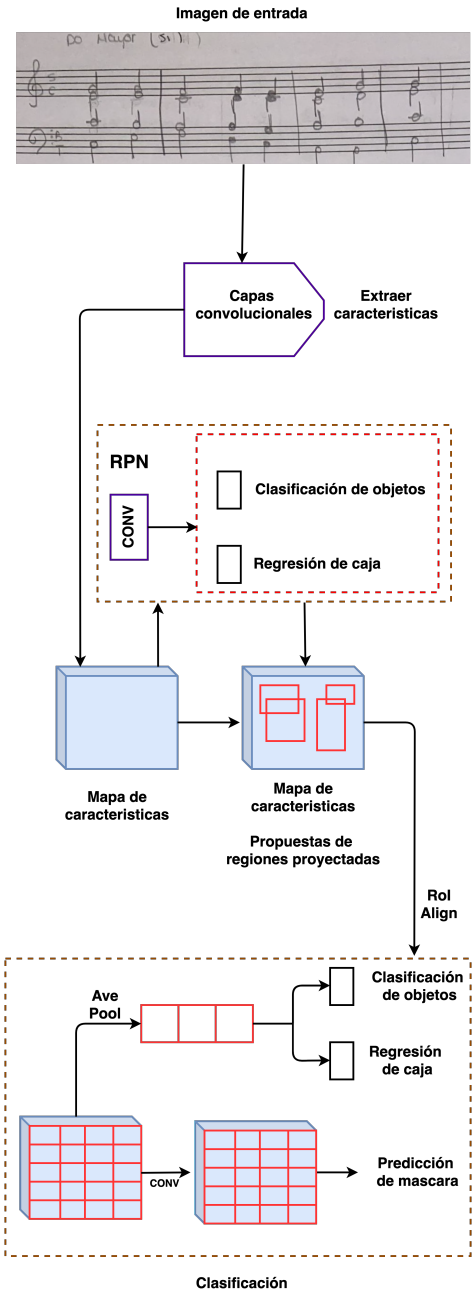


Figura 5. Arquitectura de Mask R-CNN.

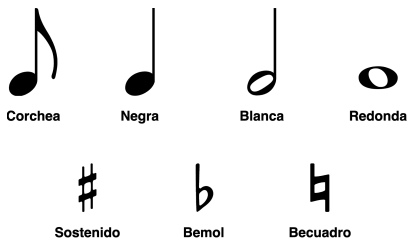


Figura 6. Símbolos utilizados en una partitura SATB.

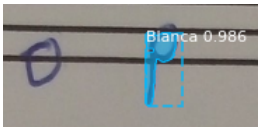


Figura 7. Símbolo correctamente identificado (derecha) y no identificado (izquierda).

i es un objeto; La variable t_i representa cuatro coordenadas parametrizadas, que son: el valor de las coordenadas horizontales y verticales del punto central en el cuadro, la anchura y la altura del cuadro, y t_i^* indica la diferencia entre el cuadro de la etiqueta verdadera y el cuadro delimitador positivo; Y s representan respectivamente las matrices binarias de la máscara de predicción y de la etiqueta verdadera.

Para este trabajo de investigación, se aplicaron las tasas de precisión (P) y recuperación (R), se pueden calcular mediante las ecuaciones 6 y 7 respectivamente, para evaluar el rendimiento de Mask R-CNN en la detección de los símbolos, donde TP es el número de casos que son positivos y se detectan como positivos, FP es el número de casos que son negativos pero que se detectan como positivos y FN es el número de casos que son positivos pero que se detectan como negativos [18].

$$P = \frac{TP}{TP + FP} \tag{6}$$

$$R = \frac{TP}{TP + FN} \tag{7}$$

De manera visual también es posible detectar los símbolos que han sido correctamente identificados, y cuales no han sido reconocidos por la CNN. Aquellos símbolos que son detectados de manera correcta están delimitados por un cuadro y una mascara que se adapta a su forma, también contienen la etiqueta de la categoría a la que pertenecen y el porcentaje de acierto con el que se ha evaluado, en la Figura 7 podemos observar estas características, también se observa un símbolo que no ha sido detectado.

IV. Implementación y resultados

Nuestra metodología la implementamos bajo el marco de desarrollo de aprendizaje profundo de TensorFlow[4] y Keras[5], programado en lenguaje Python, con una doble GPU, las cuales son NVIDIA GTX 1080 Ti y NVIDIA GTX 1080, memoria RAM de 64GB y 8TB de disco duro.

Para la investigación, se seleccionaron 100 imágenes de partituras para el entrenamiento (80 % del conjunto de entrenamiento y 20 % del conjunto de validación). Con el fin de verificar la estabilidad y confiabilidad del modelo entrenado, se utilizaron las mismas imágenes del conjunto de entrenamiento

[4]TensorFlow es una plataforma de código abierto para construir y entrenar redes neuronales, que permiten detectar y descifrar patrones análogos al aprendizaje utilizado por los humanos, https://www.tensorflow.org/?hl=es-419.

[5]Keras es una biblioteca de código abierto, tiene como objetivo de acelerar la creación de redes neuronales. Keras no funciona como un framework independiente, sino como una interfaz de uso intuitivo (API), https://keras.io/.

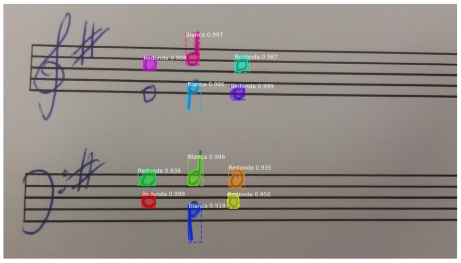


Figura 8. Predicciones de los símbolos.

para la evaluación del modelo. En este caso la arquitectura se configuro con una tasa de aprendizaje de 0,001, y se ajustó a 700 épocas de entrenamiento con un factor de ajuste de 0,90 % y 100 pasos por cada una de ellas, evaluando 1 imagen por GPU. El modelo generó las puntuaciones de las categorías, los cuadros delimitadores y las máscaras individuales de los símbolos para cada imagen de entrada. El tiempo de entrenamiento para 700 épocas fue de 11 horas, aproximadamente 1 minuto por época, y la función de pérdida del modelo alcanzó un estado de convergencia.

Los resultados de 100 imágenes de prueba mostraron que la precisión general y las tasas de recuperación fueron 95,40 % y 94,50 %, respectivamente. En la Figura 8 se presenta una detección de símbolos clasificados, junto con el porcentaje individual de precisión.

Es notable que la detección de algunos símbolos funciona mejor que otros. Las principales razones de los errores en el reconocimiento de los símbolos se deben a que el tamaño de las muestras de algunos símbolos, como son corcheas, sostenidos, bemoles y becuadros, las cuales representan el 2 %, 6 %, 5 % y 2 %, respectivamente, no son suficientes para producir resultados confiables, en comparación con blancas, redondas y negras, que tienen un promedio del 25 %, 40 % y 20 %, respectivamente. Además, varios símbolos se juzgaron mal entre sí, ya que incluso para el ojo humano eran indistinguibles, debido a la manera en que escriben los alumnos, y los resultados de la clasificación del modelo se ven afectados por errores humanos al etiquetar el conjunto de entrenamiento.

V. Discusión y trabajo futuro

Los resultados obtenidos hasta el momento nos hacen confiar en obtener una metodología que pueda ponerse a disposición de los usuarios interesados. Primero, la herramienta *Sharpmony*, actualmente disponible en Google Play Store para descarga, está siendo utilizada por más de 1800 usuarios registrados. Adicionalmente, esperamos poder incluir en *Sharpmony* la opción de captura fotográfica descrita anteriormente. No obstante, a la metodología anterior habrá que añadir la etapa que permita capturar no solo las figuras (redondas, blancas...), sino también la posición concreta sobre el pentagrama que nos permita identificar el nombre de la nota (*do, re, mi, ...*).

En pruebas preliminares estamos añadiendo a la metodología una segunda CNN, con el objetivo de identificar el tono

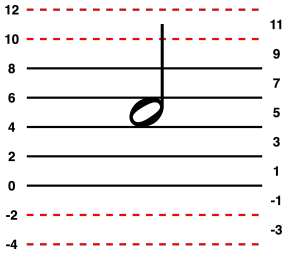


Figura 9. Clasificación de posiciones dentro en el pentagrama.

de una nota, ya que es información vital en el reconocimiento de partituras. Utilizamos la posición para representar el tono, asignando como punto de partida la quinta linea del pentagrama, identificando con números pares las lineas y con números impares los espacios. En la Figura 9, se representa de manera gráfica un ejemplo donde una nota blanca es clasificada en la posición 5. Aunque aún estamos en una etapa preliminar de análisis, en fase de entrenamiento estamos obteniendo una tasa de acierto del 85 %. Esperamos poder mejorar este valor y llegar a niveles superiores al 90 % en fase de prueba para mostrar un análisis completo de la metodología e incluir la misma en la herramienta *Sharpmony*.

VI. Conclusiones

En este trabajo hemos presentado una aproximación factible al reconocimiento de partituras escritas a mano, como una aplicación de visión por computador y aprendizaje profundo, para asistir en el proceso de enseñanza-aprendizaje de las personas que realizan estudios de música.

Partiendo de ejercicios realizados por estudiantes de armonía de conservatorios profesionales de música, hemos desarrollado una metodología que nos permite identificar las notas escritas con una tasa de acierto superior al 95 %. Además, algunas pruebas preliminares ya desarrolladas nos permiten ver que la metodología puede mantener esas tasas de reconocimiento a la hora de identificar la posiciones de las notas sobre el pentagrama, y obtener así toda la información necesaria, junto con la armadura y el compás, para poder generar la partitura en formato MusicXML, que pueda ser procesada posteriormente por cualquier software de edición musical. El objetivo final es mejorar la herramienta *Sharpmony* con las metodología aquí descrita.

El nivel de reconocimiento logrado permite dar un paso hacia adelante en el entendimiento de problemas de visión por computador, ya que una imagen puede contener información contextual acorde a un propósito específico, que para nuestro interés es el lenguaje escrito de la música.

Nuestro desarrollo ya está en operación experimental y esperamos contar con nuevas imágenes de partituras para enriquecer el reconocimiento, así como orientar a nuevos propósitos este reconocimiento.

Agradecimientos

Agradecemos el apoyo del Ministerio de Economía y Competitividad de España en el marco de los proyectos TIN2017-85727-C4-{2,4}-P y RTI2018-095180-B-I00, TIN2017-85727-C4-{2,4}-P y PID2020-115570GB-C21, de la Junta de Extremadura, Consejería de Comercio y Economía, del Fondo Europeo de Desarrollo Regional, una manera de construir Europa, bajo el proyecto IB16035 y de la Junta de Extremadura, proyecto GR15068 y GR18049. Este trabajo es resultado de la colaboración de la Universidad de Extremadura con la Universidad Autónoma Metropolitana en la Cd. de México, dentro del proyecto “Evolución artificial de descriptores estadísticos de textura de superficie para implementación en clasificación de imágenes digitales”, clave: EL006-18.

Referencias

[1] J. Redmon, S. Divvala, R. Girshick, y A. Farhadi, *You only look once: Unified, real-time object detection*, en Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 779-788.

[2] W. Liu et al., *Ssd: Single shot multibox detector*, en European conference on computer vision, 2016, pp. 21-37.

[3] T.-Y. Lin, P. Goyal, R. Girshick, K. He, y P. Dollár, *Focal loss for dense object detection*, en Proceedings of the IEEE international conference on computer vision, 2017, pp. 2980-2988.

[4] S. Ren, K. He, R. Girshick, y J. Sun, *Faster r-cnn: Towards real-time object detection with region proposal networks*, arXiv preprint arXiv:1506.01497, 2015.

[5] R. Girshick, J. Donahue, T. Darrell, y J. Malik, *Rich feature hierarchies for accurate object detection and semantic segmentation*, en Proceedings of the IEEE conference on computer vision and pattern recognition, 2014, pp. 580-587.

[6] J. Dai, Y. Li, K. He, y J. Sun, *R-fcn: Object detection via region-based fully convolutional networks*, arXiv preprint arXiv:1605.06409, 2016.

[7] F. Rossant y I. Bloch, *Robust and adaptive OMR system including fuzzy modeling, fusion of musical rules, and possible error detection*, EURASIP Journal on Advances in Signal Processing, vol. 2007, pp. 1-25, 2006.

[8] A. Pacha, J. Hajič, y J. Calvo-Zaragoza, *A baseline for general music object detection with deep learning*, Applied Sciences, vol. 8, n.o 9, p. 1488, 2018.

[9] L. Tuggener, I. Elezi, J. Schmidhuber, M. Pelillo, y T. Stadelmann, *Deepscores-a dataset for segmentation, detection and classification of tiny objects*, en 2018 24th International Conference on Pattern Recognition (ICPR), 2018, pp. 3704-3709.

[10] J. Calvo-Zaragoza, D. Rizo, y J. M. I. Quereda, *Two (Note) Heads Are Better Than One: Pen-Based Multimodal Interaction with Music Scores.*, en ISMIR, 2016, pp. 509-514.

[11] O. Ronneberger, P. Fischer, y T. Brox, *U-net: Convolutional networks for biomedical image segmentation*, en International Conference on Medical image computing and computer-assisted intervention, 2015, pp. 234-241.

[12] J. Hajič y P. Pecina, *The MUSCIMA++ dataset for handwritten optical music recognition*, en 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), 2017, vol. 1, pp. 39-46.

[13] J. Hajič Jr y P. Pecina, *Detecting noteheads in handwritten scores with convnets and bounding box regression*, arXiv preprint arXiv:1708.01806, 2017.

[14] J. Hajic Jr, M. Dorfer, G. Widmer, y P. Pecina, *Towards Full-Pipeline Handwritten OMR with Musical Symbol Detection by U-Nets.*, en ISMIR, 2018, pp. 225-232.

[15] K. He, X. Zhang, S. Ren, y J. Sun, *Deep residual learning for image recognition*, en Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770-778.

[16] L. Tuggener, I. Elezi, J. Schmidhuber, y T. Stadelmann, *Deep watershed detector for music object recognition*, arXiv preprint arXiv:1805.10548, 2018.

[17] K. He, G. Gkioxari, P. Dollár, y R. Girshick, *Mask r-cnn*, en Proceedings of the IEEE international conference on computer vision, 2017, pp. 2961-2969.

[18] Q. Yang, D. Xiao, y S. Lin, *Feeding behavior recognition for group-housed pigs with the Faster R-CNN*, Computers and Electronics in Agriculture, vol. 155, pp. 453-460, 2018.

Optimización Evolutiva de Parámetros sobre Random Forest en Entornos de Big Data

Miguel Ángel Rodríguez
Departamento de Tecnologías de la Información
Universidad de Huelva
Huelva, España

Antonio Peregrín
Centro de Estudios Avanzados en Física, Matemáticas y Computación (CEAFMC)
Instituto Interuniversitario de Investigación en Data Science and Computational Intelligence (DaSCI)
Universidad de Huelva
Huelva, España

***Resumen*—El aprendizaje automático se ha convertido en estos últimos años en una herramienta que sirve a múltiples propósitos. La complejidad de las tareas de aprendizaje son muchas veces un área de investigación en sí mismas. En este entorno altamente exigente con grandes cantidades de tuplas y características, existencia de ruido, eficiencia y calidad de predicción, resaltan algunos algoritmos, entre ellos Random Forest, por su eficacia en entornos MapReduce, llevando la escalabilidad a limites muy superiores con respecto a otros clasificadores, sin una pérdida en la calidad perceptible. Sin embargo, no esta exento de tareas de ajuste de parámetros para llegar a una calidad óptima, lo que dificulta la usabilidad del proceso de extracción de conocimiento. En este contexto la optimización evolutiva puede ayudar en la automatización de este paso. Este trabajo estudia la optimización de los parámetros de Random Forest en un entorno de Big Data distribuido mediante diferentes técnicas de optimización evolutiva.**

***Palabras clave*—Big Data; Algoritmos Genéticos; Spark; Clasificación; Optimización Evolutiva; Estrategias Evolutivas; Random Forest.**

I. Introduccion

Actualmente, enormes cantidades de datos son recogidas y almacenadas en bases de datos. El fin último de la Ciencia de los Datos es obtener información de interés en forma de modelos precisos de estas bases de datos. Es conocido que a medida que el volumen de datos crece, su manejo es más complejo, el tiempo necesario para su tratamiento se incrementa de manera exponencial, y además crece la dificultad de aprendizaje del algoritmo.

En los últimos años, la aparición del paradigma *MapReduce* [2] y el sistema de almacenamiento *HDFS* [3], han impulsado notables mejoras en la escalabilidad. No obstante, no todos los algoritmos son directamente aptos para una implementación equivalente en entornos distribuidos, por tanto, en tales casos es necesario establecer estrategias que conserven la calidad de los resultados y mantengan el tiempo de ejecución a niveles competitivos frente a otras alternativas distribuidas o paralelas.

Los algoritmos de clasificación basados en ensembles de árboles en general, y *Random Forest* [3] en particular, han demostrado una gran capacidad de extracción de conocimiento, al incluir selección de características y buen rendimiento con datos de baja calidad como el ruido, baja separabilidad de las clases y otras características inherentes a la captura de información; sin embargo, y al contrario de otras técnicas en las que suele ser difícil escalar a grandes conjuntos de datos, estos algoritmos permiten mantener un buen comportamiento predictivo y de rendimiento, tanto en la construcción como en la predicción para grandes conjuntos de datos debido a la existencia de implementaciones de los mismos en entornos de *Big Data* distribuidos basados en el paradigma *MapReduce*.

Este trabajo propone la utilización de diferentes técnicas de optimización evolutiva para mejorar la calidad de clasificación del algoritmo *Random Forest* sobre la implementación existente en la biblioteca MLlib de *Apache Spark* desde un punto de vista empírico sobre un conjunto de datos contrastado con otros clasificadores, tanto en entorno distribuido como en su versión clásica sobre un conjunto de datos monolítico.

La organización de este trabajo es la siguiente: en la segunda sección se describe la optimización de parámetros sobre clasificadores, y entre ellos, *Random Forest*. La sección 3 estudia la codificación de los parámetros en las propuestas evolutivas desarrolladas y la justificación de los operadores evolutivos. Finalmente, la sección 4 analiza los resultados de precisión y tiempo sobre una experimentación preliminar de diferentes implementaciones de clasificadores sobre el conjunto de datos de referencia.

II. Optimización de parametros en RAndom Forest

La optimización de modelos es también conocida como selección de modelos o por su termino en inglés *hyperparameter tuning*. Podemos definir este objetivo como el de encontrar el mejor conjunto de parámetros para un proceso de aprendizaje. Esta tarea ha tomado en los últimos años una gran relevancia debido al creciente numero de métodos que siguen un proceso guiado por datos. La automatización de tareas de adquisición de datos ha llevado al uso de técnicas que sean capaces de generar modelos con muy poca transformación y adaptación de los datos originales. Una de las técnicas que más ha crecido es el uso de redes neuronales en aprendizaje, *Deep Learnig,* que al depender en gran medida de una correcta elección de sus parámetros de construcción [4] ha impulsado la investigación en optimización de parámetros.

Esta tarea en sí resulta de una complejidad creciente debido a que la mayor parte de sistemas de optimización de modelos aplican una meta-validación como medida para estimar la bondad de un modelo. Esto es, realizan una serie de ejecuciones con un conjunto de datos representativos del problema, y un conjunto de parámetros conocidos para comparar su precisión con el resultado esperado.

Un enfoque muy común y a menudo recomendado es el uso de productos cartesianos (*grid*) de valores, en el que se sitúa una distribución homogénea sobre el espacio de posibles parámetros y se evalúa cada combinación resultante; las combinaciones con mejor valor objetivo se utilizan en producción. Este método tiene baja escalabilidad, pues crece exponencialmente con el aumento del número de parámetros y la amplitud de los rangos de estos. Asimismo, puede no garantizar la mejor combinación si los parámetros no siguen un comportamiento lineal o el espaciado del producto cartesiano no es suficiente para explorar el espacio de un parámetro. En este sentido, en [5] se propone un sistema de paralelización de parámetros, que permite mejorar la sistematización y ejecución de diferentes grupos de valores.

Otra circunstancia que limita la escalabilidad es la relativa a la cantidad de datos (cardinalidad o dimensionalidad) a procesar en la evaluación. Una estrategia para trabajar en baja cardinalidad/dimensionalidad sobre un conjunto de datos no abarcable es la realización de un muestreo para redimensionar el conjunto de datos con el que realizar la evaluación. Sin embargo, esta técnica puede derivar en una perdida importante de precisión del clasificador y por tanto falsear los resultados del mismo. En este punto toman importancia las plataformas de *Big Data*, que permiten evaluar algunos clasificadores con el conjunto completo de datos sin perder calidad en el modelo.

El uso de algoritmos evolutivos para el ajuste de parámetros de modelos de clasificación es una solución ya probada de mejora respecto a la cuadricula de valores [6][7], sin embargo, según nuestro conocimiento, no existen propuestas que puedan realizar esta tarea sobre conjuntos grandes de datos. Por este motivo, la elección de *Random Forest* es idónea como ejemplo para la optimización evolutiva de parámetros escalable y sin pérdidas, al tener un reducido numero de parámetros y una ejecución equivalente a su homónimo no distribuido.

Random Forest es un algoritmo basado en conjuntos de árboles de decisión, los cuales son modelos de clasificación débiles, que combinando los resultados de los mismos permiten obtener un modelo robusto.

La construcción de los bosques de arboles en *Random Forest* es un proceso eficiente respecto a la cantidad de tuplas/características. Se apoya en el concepto de discriminación estocástica [8] mediante la repartición de parte de los datos en bolsas (*bagging)* y la selección de un número de características aleatorias [9], con el objeto de construir un árbol en cada una de estas bolsas, posteriormente evaluados mediante un método de *Out-of-bag error,* para determinar su peso en el votado final de la clase o aportación a la regresión. La Figura 1 muestra el esquema general de funcionamiento de *ensembles de bagging* con votado.

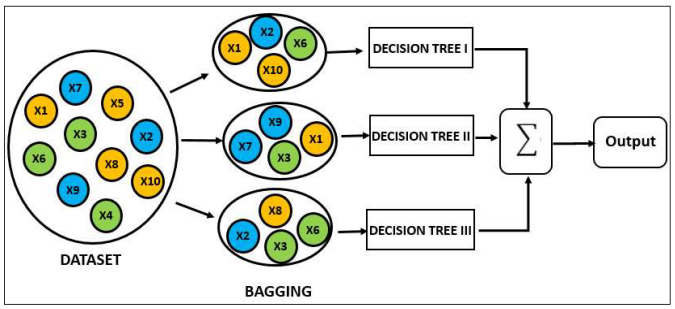


Fig. 1. Clasificador Ensemble Bagging

Los principales parámetros de construcción de este algoritmo son los siguientes:

- *Profundidad (maxDepth).* Es el nivel máximo de profundidad de todos los árboles del bosque; está muy relacionado con el número de nodos que puede tener cada árbol; un solo nivel de profundidad representa un nodo raíz con dos nodos hojas, y el nivel cero representa un solo nodo hoja. Un valor alto en este parámetro podría aumentar la precisión en el resultado de la clasificación, aunque previsiblemente con un mayor coste computacional.
- *Árboles.* Es el número de árboles de decisión que se emplean en el bosque. Un valor elevado de este parámetro no garantiza un buen resultado en el clasificador, y respecto al coste computacional, representa uno de los factores más relevantes.
- *Bins o Máximo de puntos.* Determina el número máximo de puntos para la división de las características en cada nodo. Un mayor número podría suponer un mayor coste computacional.
- *Características.* Es el número de características tomadas para hacer las divisiones en cada nodo de cada árbol del bosque.
- *Muestreo.* Fracción de los datos de entrenamiento, que utiliza cada árbol de decisión para construir el modelo. Valores altos podrían incrementar el rendimiento del modelo, a costa de un sobreajuste en el aprendizaje.

El sobreajuste es un termino a menudo relacionado con los *ensembles* de árboles [10]. Algunas de las relaciones conocidas de los parámetros con esta dificultad son:

- Utilizar la máxima profundidad disponible para el algoritmo, ya que en este caso los árboles no van a generalizar adecuadamente, prediciendo erróneamente los datos de *test*.
- Utilizar todas las características del conjunto de datos.
- Establecer el valor de muestreo a un valor elevado, ya que reduce la validación de los árboles.

- Utilizar un número muy elevado de árboles.

III. Optimización Evolutiva

El uso de una búsqueda exhaustiva sobre un conjunto de datos puede llegar a ser elevado. El número de combinaciones a explorar viene dado por la ecuación:

Profundidad*Arboles *Bins*Características*Muestreo (1)

Esta búsqueda tiene orden exponencial, lo que justifica el uso de técnicas metaheurísticas de optimización para reducir el número de ejecuciones necesarias.

Proponemos el uso de dos técnicas evolutivas bien conocidas en el entorno de la optimización como son los algoritmos genéticos y las estrategias evolutivas. Ambas estrategias utilizan la representación mediante un cromosoma, guiado por el valor de la función de evaluación. El cromosoma está compuesto por los parámetros del algoritmo de clasificación: profundidad, arboles, puntos de división de la variable (*Bins*), número de atributos y ratio de muestreo. Los parámetros se codifican como un vector de números naturales con límites superiores e inferiores.

A. Algoritmo Genético

En este trabajo se ha implementado el cruce *Parent Centered* BLX-α (PCBLX-ALPHA), con modelo estacionario, selección aleatoria y reemplazo por *crowding* determinístico modificado para sólo aceptar hijos con mejora sobre el valor de alguno de los padres. Dada la elevada presión selectiva del operador de *crowding*, se ha utilizado un numero superior de descendientes en cada cruce, ya que valores bajos de este parámetro tienen una baja convergencia y mayores valores afectan al rendimiento del algoritmo. Utilizamos el operador de PCBLX-ALPHA [11] frente al cruce BLX-ALPHA [12] utilizado originariamente en el modelo CHC. Según lo mostrado en [11] el cruce PCBLX-ALPHA presenta mejores resultados que el cruce BLX-ALPHA.

El operador de mutación usa una distribución gaussiana con un parámetro de apertura α sobre el rango del valor mínimo-máximo de cada parámetro. El valor de la función de evaluación será la medida de precisión del clasificador *Random Forest* sobre la biblioteca MLlib de *Apache Spark* con un conjunto de datos de referencia.

B. Estrategias Evolutivas

Las Estrategias Evolutivas fueron desarrolladas [13] para resolver problemas hidrodinámicos de alto grado de complejidad. La versión original (1+1)-EE usa un padre y con él se genera un hijo. El algoritmo de Estrategia Evolutiva (1+1), aunque se podría calificar de búsqueda local, sin embargo, tiene una inspiración genética. El cromosoma contiene dos partes: en la primera se encuentran los coeficientes de mutación y en la segunda los genes numéricos que se van a optimizar.

Al igual que el Algoritmo Genético, utiliza como función de coste la precisión obtenida en la clasificación realizada sobre los datos por el algoritmo *Random Forest*.

El operador que obtiene los descendientes de la solución actual es una mutación gaussiana, con media cero y desviación sigma. Este valor no permanece constante, pues se va actualizando según la tasa de éxitos; con un número de aciertos bajo disminuye el paso, y realiza mutaciones menos frecuentes pero mayores; sin embargo, con un número de aciertos alto aumenta el paso conmutaciones más frecuentes, pero menos importantes (Figura 2).

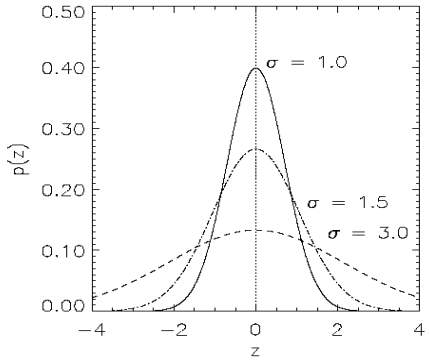


Fig. 2. Efecto de la auto-adaptación de sigma

La versión poblacional de la Estrategia Evolutiva, se denomina (μ+λ)-EEs o (μ,λ)-EEs, con λ padres y μ hijos. Estas dos variantes se diferencian en la pertenencia de los padres al conjunto de soluciones que pueden ser seleccionados para formar la siguiente generación.

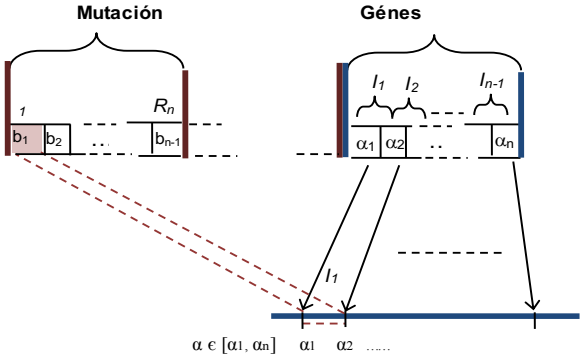


Fig. 3. Cromosoma doble de Mutaciones-Genes

En este trabajo se ha optado por la versión (μ+λ)-EEs, que asegura una mayor presión selectiva al mantener las mejores soluciones encontradas hasta el momento.

Las Estrategias Evolutivas generan hijos mediante la selección aleatoria de dos padres, en la que se produce una operación de cruce que determina la mutación gaussiana que se le aplicará al gen numérico. En la Figura 3 se muestra como el cromosoma completo esta formado por dos partes, donde una vez seleccionado el gen de uno de los padres, se le aplicará a

ese gen la mutación correspondiente al gen heredado de la parte del cromosoma correspondiente a la mutación. De esta forma se produce una auto-adaptación de los valores de mutación al tener más probabilidades de supervivencia en aquellos que generan individuos de alto valor.

IV. Estudio Experimental

Esta sección describe una experimentación tentativa sobre un conjunto de datos representativo y su comparación con la misma versión del algoritmo de referencia en *Apache Spark* (entorno distribuido) y WEKA (entorno no distribuido). Se estimará la calidad del clasificador generado en base a la precisión (media geométrica).

En cuanto a los datos utilizados en la experimentación, se ha seleccionado el dataset de *Electroencefalograma* para detectar el estado del ojo abierto o cerrado (*EEG Eye State*). Éste es un conjunto de datos bien contrastado para *Random Forest* y otras técnicas de clasificación [14], disponible en el repositorio UCI [15] . Si bien no es un dataset considerado de *Big Data* (14980 instancias y 15 características), se ha utilizado en esta experimentación preliminar para validar tiempos y calidades respecto a la ejecución en entornos no distribuidos y por tanto sin la capacidad de escalar a dataset de mayor cardinalidad o dimensionalidad. Por tanto, nos referimos a entornos de *Big Data* por el entorno de ejecución y la posibilidad de escalabilidad ilimitada, no por el uso de un dataset de dichas dimensiones.

A. Entorno de experimentación

El entorno de experimentación se basa en un *clúster* con *Apache Spark* compuesto por 13 nodos. Cada nodo es una máquina virtual sobre un servidor basado en memoria común y multiprocesador: 192GB de memoria RAM, 40 núcleos con *hyperthreading* y sistema operativo CentOS 7 de 64 bits. Todo el desarrollo se ha realizado en Java. Las características de las máquinas virtuales son:

Nodo maestro:

- Sistema operativo CentOS 7 64 bits, y 8 núcleos.
- 16 GB de memoria RAM.
- 20 GB de almacenamiento en disco configurado en modo dinámico.

Nodos esclavos:

- Sistema operativo CentOS 7 64 bits, y 4 núcleos.
- 12 GB de memoria RAM.
- 20 GB de almacenamiento en disco configurado en modo dinámico.

Las tablas muestran valores medios de ejecuciones con 10 semillas y 5x2 particiones de datos.

B. Parámetros de ejecución y resultados para EEG Eye State

Para el Algoritmo Genético tenemos las siguientes configuraciones de parámetros:

- *Particiones:* numero de particiones del *clúster*. Los valores posibles son 1, 6 y 12.
- *Población:* número de individuos de la población: 10, 15, 20 individuos.
- *Hijos o numero de descendientes por cruce:* 2-4 hijos por cruce.

Se han realizado para cada algoritmo una serie de ejecuciones con todas las combinaciones para cada parámetro (27) en las mismas condiciones que el resto de la experimentación: 10 semillas y validación cruzada 5x2. El resultado promedio de precisión y tiempo de ejecución en segundos es el mostrado en la Tabla I.

TABLA I. Parámetros del Algoritmo Genético

Parámetro	Valor	Precisión	Tiempo
Particiones	1	92,15233489	151,49
	6	92,14324283	170,39
	12	92,07659816	174,74
Población	10	92,06686724	150,99
	15	92,06842256	147,27
	20	92,23688604	198,35
Hijos	2	92,00570593	160,86
	3	92,20768513	208,04
	4	92,15878479	127,71

La Tabla I, muestra los valores obtenidos en el Algoritmo Genético. Cada fila es el resultado medio de la ejecución con la variación del resto de parámetros. Por ejemplo, el resultado con partición igual a 1, es la media de las ejecuciones con los valores de población (10,15,20) y de número de hijos (2, 3,4). Para este dataset, el número de particiones y la población no tienen efecto sobre la calidad de la solución por lo que la opción mas eficiente es la correspondiente a los valores de partición a 1, tamaño de población a 15 y numero de hijos a 4.

TABLA II. Resultados Algoritmo Genético

Modelo		Random Forest					A. Genético		
Precisión	*Tiempo*	*P*	*A*	*B*	*C*	*M*	*P*	*T*	*H*
92,5899	88,01	20	93	65	8	100	1	15	4
92,5899	88,83	20	93	65	8	100	6	15	4
92,5758	106,97	26	105	48	7	97	1	15	3
92,5758	109,01	26	105	48	7	97	6	15	3
92,5758	197,04	26	105	48	7	97	12	15	3

En el desglose de la Tabla II se muestran calidad y tiempo según los parámetros del Algoritmo Genético (P, T y H por las particiones, el tamaño de la población y el número de hijos del cruce respectivamente) y el cromosoma optimizado (P, A, B, C y M por la profundidad máxima, el número de árboles (*Bins*), el número de características y el *ratio* de muestreo). Se puede observar que la combinación más eficiente alcanza la misma precisión en un tiempo un 44% de la opción mas costosa.

La Estrategia Evolutiva (1+1) tiene los parámetros iniciales:

- *Particiones:* numero de particiones del *clúster*. Los valores posibles son 1, 6 y 12.

- *Sigma:* desviación estándar inicial que aplica el algoritmo en la mutación gaussiana.
- *Alfa:* tamaño del paso tomado para la mutación gaussiana, el porcentaje sobre el rango de valores de cada gen del cromosoma.

Para las ejecuciones con todas las combinaciones de los valores de estos tres parámetros, se han obtenido los resultados (Tabla III), y posteriormente se han analizado estos resultados para analizar su impacto sobre la calidad de los modelos obtenidos del algoritmo *Random Forest* y al tiempo de ejecución en segundos de los mismos (Tabla IV). Se puede observar en esta tabla que no hay una diferencia notable en la calidad o el tiempo de ejecución del algoritmo respecto de los valores de los parámetros.

TABLA III. PARÁMETROS ESTRATEGIA EVOLUTIVA (1+1)

Parámetro	Valor	Precisión	Tiempo
Particiones	1	91,82827037	164,73
	6	91,94958832	176,63
	12	92,03842031	179,51
Sigma	1,0	92,01664893	163,09
	2,0	91,90121024	172,75
	3,0	91,89841988	185,03
Alfa	10%	91,80414165	171,80
	20%	92,06087091	159,83
	40%	91,95126645	189,25

TABLA IV. RESULTADOS ESTATREGIA EVOLUTIVA (1+1)

Modelo		Random Forest					(1+1)-EE		
Precisión	***Tiempo***	***P***	***A***	***B***	***C***	***M***	***P***	***T***	***H***
92,63310	222,44	26	136	47	6	92	1	2,0	20
92,63310	224,51	26	136	47	6	92	6	2,0	20
92,63310	225,50	26	136	47	6	92	12	2,0	20
92,63310	222,44	26	136	47	6	92	1	2,0	20
92,63310	224,51	26	136	47	6	92	6	2,0	20

Para la Estrategia Evolutiva (μ+λ)-EEs, se ha seguido el mismo esquema de experimentación, un estudio del impacto de los parámetros del algoritmo, seguido de un desglose se los principales resultados obtenidos por la misma. Las descripciones y conjuntos de valores son los mismos que para el algoritmo. genético.

Se han realizado ejecuciones con las combinaciones de los valores de estos tres parámetros para estimar los mejores valores para la experimentación (Tabla V). En la Tabla V se evidencia que las Estrategias Evolutivas (μ+λ)-EEs no son sensibles a estos parámetros, adquiriendo prácticamente la misma precisión en tiempos similares. La Tabla VI contiene los cinco mejores resultados del experimento para el conjunto de datos *EEG Eye State*. Aunque en media no se aprecia diferencia el número de individuos de la población parece tener mejor comportamiento con valores bajos de la misma. Se ha tomado un valor de μ+λ según el siguiente esquema relativo a la población inicial: con μ = población, λ tomará valores dependientes del número de hijos del cruce; para hijos = 2 será igual a μ; para hijos = 3 , será 1,5*μ y finalmente para hijos = 4 corresponderá a 2*μ.

Es interesante remarcar que los Algoritmos Genéticos han obtenido soluciones con menor profundidad, menor número de arboles y mayor número de características de una manera consistente en la experimentación para precisiones finales similares sin que haya un motivo aparente en los algoritmos que induzcan al crecimiento o decrecimiento de una variable en particular. La explicación mas plausible es que las Estrategias Evolutivas caen en un mínimo con valores mayores de profundidad y número de árboles pues son una forma de ganar en precisión a costa de un posible sobre aprendizaje, que penaliza finalmente en la fase de *test* a las Estrategias Evolutivas.

TABLA V. PARÁMETROS ESTRATEGIA EVOLUTIVA (μ+λ)-EES

Parámetro	Valor	Precisión	Tiempo
Particiones	1	92,10794304	164,56
	6	92,14232812	156,78
	12	92,11348022	141,19
Población	10	92,14333944	155,19
	15	92,11506412	155,66
	20	92,10534782	156,66
Hijos	2	92,12919329	137,16
	3	92,14221514	174,91
	4	92,09234295	153,71

TABLA VI. RESULTADOS ESTRATEGIA EVOLUTIVA (μ+λ)-EES

Modelo		Random Forest					(μ+λ)-EE		
Precisión	***Tiempo***	***P***	***A***	***B***	***C***	***M***	***P***	***T***	***H***
92,36238	185,02	23	129	29	6	97	1	10	4
92,36238	188,16	23	129	29	6	97	6	10	4
92,28377	164,04	24	136	53	8	95	6	15	4
92,28377	142,55	24	136	53	8	95	12	15	4
92,28078	133,37	22	104	67	6	87	1	10	3

TABLA VII. RESULTADOS CLASIFICADORES DISTRIBUIDOS SPARK

Algoritmo	Precisión	Tiempo
Random Forest	89,90	75,188
Boosted Trees	72,26	20,794
Logistic Regression	60,79	22,189
SVM	54,90	13,517
Naïve Bayes	45,37	07,264

TABLA VIII. RESULTADOS CLASIFICADORES WEKA

Algoritmo	Precisión
Random Forest	89,5
Bagging	87,0
Logistic Regression	84,4

Para analizar la calidad de clasificadores con implementación en *Apache Spark* y en WEKA, se ha realizado una comparativa con otros algoritmos de clasificación sobre el mismo *dataset* con los parámetros aconsejados en la biblioteca MLlib. Los resultados se muestran en la Tabla VII. Finalmente, en la Tabla VIII se muestran los resultados obtenidos por [14] en clasificadores similares mediante el uso de WEKA [16] con los parámetros por defecto. Se puede comprobar que los valores obtenidos en *Random Forest* por ambos conjuntos de clasificadores obtienen una puntuación casi idéntica, pero inferior a la obtenida con la optimización paramétrica de los algoritmos propuestos. El comportamiento de los *ensembles* de arboles en entornos de programación distribuida es el mismo que en su equivalente no distribuido (WEKA) debido a que el particionamiento de datos no modifica la estructura algorítmica del mismo. Sin embargo, otras implementaciones como la

regresión o las maquinas de vector soporte o Naive Bayes tienen un comportamiento inferior en la versión distribuida.

V. Conclusiones

Este trabajo presenta una propuesta de optimización evolutiva de los parámetros del clasificador *Random Forest* en entornos de alta escalabilidad con *Apache Spark*, lo que permite la posibilidad de ejecutar la optimización de parámetros con todo el conjunto de datos y la escalabilidad del proceso de tuning sin perdida de calidad al usar una implementación equivalente a la existente en baja cardinalidad.

Aunque la optimización evolutiva lleva acompañada una carga de parámetros adicionales al funcionamiento del algoritmo, en esta experimentación se ha seguido una pauta que permite reducir razonablemente la casuística a parámetros con una considerable precisión y un menor tiempo de ejecución.

En cuanto al comportamiento comparado de las tres técnicas expuestas, cabe destacar que todas tienen un comportamiento similar en cuanto a precisión alcanzada, aunque el Algoritmo Genético alcanza la mejor precisión y los tiempos más reducidos (Tablas II, IV y VI), siendo además el modelo generado más compacto. Podemos intuir que ambas Estrategias Evolutivas promueven un mayor número de arboles y profundidad que le penaliza en forma de sobre aprendizaje y baja la puntuación en la fase de prueba.

Tras esta experimentación preliminar este trabajo plantea como posible continuación comparar la optimización con otras técnicas metaheurísticas ampliamente utilizadas en la optimización con codificación real como PSO (*Particle Swarm Optimization*) o DE (*Differecial Evolution*) en entornos distribuidos. En PSO [17] ha obtenido resultados sobre una aplicación no distribuida en *Random Forest* y para DE [18] analiza la capacidad de esta metaheurísticas en tuning de parámetros, pero no para el caso de *Random Forest*.

Agradecimientos

Este trabajo ha sido financiado por el Ministerio de Educación y Ciencia dentro del Proyecto TIN2017-89517-P.

Referencias

[1] J. Dean, S. Ghemawat, “MapReduce: simplified data processing on large clusters”. Commun ACM 51, pp. 107–113, 2008.

[2] T. White, Hadoop: The definitive guide, 1st ed.Sebastopol, CA: O’Reilly , 2009.

[3] L. Breiman, “Random forests”. Machine learning, 45(1), 5-32, 2001.

[4] X. Zhang, X. Chen, L. Yao, C. Ge, and M. Dong. “Deep neural network hyperparameter optimization with orthogonal array tuning”. In International Conference on Neural Information Processing ,pp. 287-295 Springer, Cham, December 2019.

[5] L. Li, K. Jamieson, A. Rostamizadeh, E. Gonina, M. Hardt, B. Recht and A. Talwalkar, “A system for massively parallel hyperparameter tuning”, preprint arXiv:1810.05934, 2018.

[6] J. Safarik, J. Jalowiczor, E. Gresak and J. Rozhon, “Genetic algorithm for automatic tuning of neural network hyperparameters”, in *Autonomous Systems: Sensors, Vehicles, Security, and the Internet of Everything*, vol. 10643, p. 106430Q, International Society for Optics and Photonics, May 2018.

[7] M. Daviran, A. Maghsoudi, R. Ghezelbash and B. Pradhan, ”A new strategy for spatial predictive mapping of mineral prospectivity: automated hyperparameter tuning of random forest approach” ,Computers & Geosciences, vol. 148, p. 104688, 2021.

[8] E. Kleinberg, "Stochastic discrimination". Annals of Mathematics and Artificial Intelligence, vol 1 (1–4), pp. 207–239, 1990.

[9] T. K. Ho, “Random decision forests”, Proceedings of the 3rd International Conference on Document Analysis and Recognition, Montreal, QC, 14–16 August 1995. pp. 278–282.

[10] P. Probst, M. N. Wright and A. L. Boulesteix, ”Hyperparameters and tuning strategies for random forest”, Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, vol. 9(3), p. e1301, 2019.

[11] H. Herrera,M. Lozano and A. Sánchez, “A taxonomy for the crossover operator for real-coded genetic algorithms: An experimental study”, International Journal of Intelligent Systems, vol. 18. pp. 309-338, 2003.

[12] L. Eshelman, “Real-coded genetic algorithms and interval schemata”, Foundations of Genetic Algorithms vol. 2, L.D, Whitley ed., pp. 187-202, 1993.

[13] I. Rechenberg, “Evolutionstrategie: optimierung technischer systeme nach prinzipien der biologischen evolution”, Frommann-Holzboog Verlag, 1973.

[14] M. Sahu, N. K. Nagwani, S. Verma and S. Shirke, “Performance evaluation of different classifier for eye state prediction using EEG signal”, International Journal of Knowledge Engineering, vol 1(2), pp. 141-145, 2015.

[15] A. Asuncion and D. Newman, UCI machine learning repository, 2007.

[16] I. H. Witten, E. Frank, L. E. Trigg, M. A. Hall, G. Holmes and S. J. Cunningham, “Weka: practical machine learning tools and techniques with Java implementations”, 1999.

[17] B. T. Pham, C. Qi, L. S. Ho. T. Nguyen-Thoi, N. Al-Ansari, M. Nguyen, and I. Prakash, “A novel hybrid soft computing model using random forest and particle swarm optimization for estimation of undrained shear strength of soil”, Sustainability, vol 12(6), pp. 2218.

[18] M. Schmidt, S. Safarani, J. Gastinger, T. Jacobs, S. Nicolas and A. Schülke, ”On the performance of differential evolution for hyperparameter tuning”, in *2019 International Joint Conference on Neural Networks (IJCNN)* pp. 1-8. IEEE, July 2019.

Un estudio de los diferentes criterios aplicables al problema de resumen extractivo multi-documento

Jesus M. Sanchez-Gomez
Depto. Tecn. Computadores y Comunicaciones
Universidad de Extremadura
Cáceres, España
jmsanchezgomez@unex.es

Miguel A. Vega-Rodríguez
Depto. Tecn. Computadores y Comunicaciones
Universidad de Extremadura
Cáceres, España
mavega@unex.es

Carlos J. Pérez
Depto. Matemáticas
Universidad de Extremadura
Cáceres, España
carper@unex.es

***Resumen*—Realizar resúmenes automáticos de textos es un tema de interés en muchos ámbitos. El problema de resumen extractivo multi-documento se puede formular como un problema de optimización que implica varios criterios. La cobertura del contenido y la reducción de la redundancia han sido los más utilizados, pero también existen otros como la relevancia y la coherencia que no han sido estudiados simultáneamente. Por este motivo, se ha realizado un estudio de los diferentes criterios aplicables a este problema. Se han analizado todas las posibles combinaciones de criterios desde el punto de vista de la optimización multi-objetivo. Los experimentos se han basado en los conjuntos de datos de DUC, y las combinaciones se han evaluado con las métricas ROUGE. Se ha demostrado que la reducción de la redundancia es el criterio más relevante, mientras que la coherencia es el menos importante. Además, el uso de la cobertura del contenido, la reducción de la redundancia y la relevancia ha proporcionado los mejores resultados en relación a su tiempo de ejecución.**

***Palabras clave*—Cobertura del contenido, Coherencia, Optimización multi-objetivo, Reducción de la redundancia, Relevancia, Resumen extractivo.**

I. Introducción

Hoy en día existe una gran cantidad de información en Internet, que crece continuamente. Este volumen de datos dificulta a los usuarios poder extraer información acerca de un tema concreto. Mediante las herramientas de minería de texto es posible realizar esta tarea [1], ya que son capaces de generar un resumen de forma automática a partir de una colección de información textual [2].

Los resúmenes automáticos pueden ser de varios tipos. Se pueden clasificar en resúmenes mono-documento, en los que se reduce la información de un único documento, y en multi-documento, en los que la información seleccionada proviene de varios documentos [3]. También pueden clasificarse en abstractivos o extractivos [4]. Un resumen abstractivo puede estar formado por texto que no existe en la fuente original, mientras que un resumen extractivo selecciona oraciones de la fuente. Por otra parte, el problema de resumen extractivo multi-documento se puede tratar como un problema de optimización mono-objetivo o multi-objetivo. En mono-objetivo solo se optimiza una única función objetivo [5], que puede incluir más de un criterio mediante la ponderación de sus pesos. Por el contrario, en multi-objetivo es posible optimizar varias funciones objetivo simultáneamente [6]. Además, los resultados obtenidos con una aproximación multi-objetivo mejoran a los obtenidos con el enfoque mono-objetivo [6].

La generación de un resumen extractivo multi-documento implica una serie de criterios a optimizar. En la mayoría de trabajos, estos criterios han sido la cobertura del contenido y la reducción de la redundancia (ver p. ej. [5], [6] y [7]). En menor medida se han utilizado los criterios de la relevancia [8] y la coherencia [9]. Por lo tanto, es interesante analizar el rendimiento de las distintas combinaciones que se pueden formar con estos criterios.

En este trabajo se ha abordado el problema de resumen extractivo multi-documento mediante un enfoque de optimización multi-objetivo. Se han estudiado e implementado como funciones objetivo la cobertura del contenido, la reducción de la redundancia, la coherencia y la relevancia. Los experimentos de las distintas combinaciones de estas funciones objetivo han sido realizados con los conjuntos de datos de DUC (*Document Understanding Conferences*) [10] y han sido evaluadas con las métricas ROUGE (*Recall-Oriented Understudy for Gisting Evaluation*) [11] para su posterior comparación.

II. Trabajo relacionado

Los trabajos relacionados con el problema de resumen extractivo multi-documento se presentan a continuación. En primer lugar se revisan los trabajos en los que se han empleado dos criterios. Todos ellos utilizaron tanto la cobertura del contenido como la reducción de la redundancia. En [5] se propuso un algoritmo adaptativo de evolución diferencial (DE). En [12] se desarrolló un algoritmo DE binario. El algoritmo DE también se aplicó en [13]. El algoritmo DE adaptativo con una novedosa estrategia de mutación se desarrolló en [14]. Por último, en [15] se utilizó un enfoque metaheurístico basado en la optimización del olfato de tiburón (SSO). Estos trabajos emplearon enfoques de optimización mono-objetivo, ponderando en una única función objetivo los criterios de la cobertura del contenido y de la reducción de la redundancia. Solo dos trabajos han empleado optimización multi-objetivo con estos dos criterios. En [7] se propuso un algoritmo genético de ordenación no dominada II (NSGA-II), y en [6] el problema se revolvió usando el algoritmo de colonia de abejas artificiales multi-objetivo (MOABC).

En segundo lugar, se revisan los trabajos que utilizaron tres criterios. Los criterios de la cobertura del contenido y de la reducción de la redundancia se mantuvieron, cambiando el tercero. En [8] se empleó el criterio de la relevancia, implementando un algoritmo DE auto-adaptativo. Por otro lado, en [9] se consideró el criterio de la coherencia, usando también un algoritmo DE auto-adaptativo. Ambos trabajos usaron enfoques de optimización mono-objetivo, por lo que los tres criterios tuvieron que ser ponderados.

III. Definición del problema

En esta sección el problema de resumen extractivo multi-documento se formula como un problema de optimización. En este campo, los métodos basados en vectores de palabras son los más utilizados. En ellos, cada oración se representa como un vector de palabras, y para medir la similitud entre oraciones se utiliza un criterio, siendo el más usado la similitud coseno.

La colección de N documentos $D = \{d_1, d_2, \ldots, d_N\}$ también puede representarse como el conjunto de las n oraciones de la colección, esto es, $D = \{s_1, s_2, \ldots, s_n\}$. Se considera también el conjunto $T = \{t_1, t_2, \ldots, t_m\}$ que contiene los m términos distintos en D. De esta forma, cada oración s_i puede representarse como un vector de términos, $s_i = (w_{i1}, w_{i2}, \ldots, w_{im})$, $i = 1, 2, \ldots, n$, donde cada componente w_{ik} es el peso del término t_k en la oración s_i. Este peso se calcula empleando el esquema tf_isf (*term-frequency inverse-sentence-frequency*) [16]:

$$w_{ik} = tf_{ik} \cdot log(n/n_k), \tag{1}$$

siendo tf_{ik} el número de veces que aparece el término t_k en la oración s_i. La expresión isf, representada como $log(n/n_k)$, cuenta las n_k oraciones que contienen el término t_k.

Ahora es posible definir la similitud coseno, medida como la semejanza entre dos oraciones $s_i = (w_{i1}, w_{i2}, \ldots, w_{im})$ y $s_j = (w_{j1}, w_{j2}, \ldots, w_{jm})$ de la siguiente manera:

$$sim(s_i, s_j) = \frac{\sum_{k=1}^{m} w_{ik} w_{jk}}{\sqrt{\sum_{k=1}^{m} w_{ik}^2 \cdot \sum_{k=1}^{m} w_{jk}^2}}, \; i, j = 1, 2, \ldots, n. \tag{2}$$

Además, el contenido principal de D, o centro de la colección, puede representarse como un vector, $O = (o_1, o_2, \ldots, o_m)$, y cada componente o_k se calcula como la media de los pesos de cada término t_k en todas las oraciones de la colección:

$$o_k = \frac{1}{n} \sum_{i=1}^{n} w_{ik}, \; k = 1, 2, \ldots, m. \tag{3}$$

Por último, la representación de la solución se expresa como $X = (x_1, x_2, \ldots, x_n)$, donde x_i es una variable binaria que tiene en cuenta la presencia o ausencia ($x_i = 1$ o $x_i = 0$) de la oración s_i en el resumen S.

III-A. Cobertura del contenido

La primera función objetivo es la cobertura del contenido. Según esta, el resumen debe cubrir el tema principal de la colección de documentos mediante la inclusión de las oraciones adecuadas [6], y se expresa como la similitud entre cada oración del resumen $s_i \in S$ y el centro de la colección (representado por O). La Ecuación (4) define esta función, para la que un valor mayor corresponde con una mayor similitud entre las oraciones del resumen y el tema principal de la colección, por lo que debe maximizarse.

$$\Phi_{CC}(X) = \sum_{i=1}^{n} sim(s_i, O) \cdot x_i. \tag{4}$$

III-B. Reducción de la redundancia

La segunda función objetivo es la reducción de la redundancia. Este criterio establece que el resumen debe evitar la repetición de oraciones parecidas entre sí [6]. Esto es, la similitud entre cada par de oraciones $s_i, s_j \in S$ debe minimizarse, lo que es equivalente a maximizar la reducción de la redundancia. La Ecuación (5) contiene la función objetivo a maximizar, que necesita una segunda variable binaria y_{ij} que considere la presencia o ausencia simultánea ($y_{ij} = 1$ o $y_{ij} = 0$) del par de oraciones s_i y s_j en el resumen S, esto es:

$$\Phi_{RR}(X) = \frac{1}{\left(\sum_{i=1}^{n-1} \sum_{j=i+1}^{n} sim(s_i, s_j) \cdot y_{ij}\right) \cdot \sum_{i=1}^{n} x_i}. \tag{5}$$

III-C. Relevancia

La tercera función objetivo se refiere a la relevancia. Este criterio trata de que el resumen contenga la información más importante incluyendo las oraciones que son relevantes para el contenido principal de la colección [8], definiéndose como el producto de la similitud entre el resumen S y el centro de la colección O y la similitud entre el resumen S y cada oración $s_i \in S$ del propio resumen. El resumen S se representa como un vector de m pesos como $S^* = (u_1, u_2, \ldots, u_m)$, donde cada elemento u_k se calcula como:

$$u_k = \frac{1}{n_S} \sum_{i=1}^{n} w_{ik} \cdot x_i, \tag{6}$$

siendo n_S el número de oraciones en S. Por tanto, el vector de pesos S^* del resumen S se calcula como el promedio de los pesos correspondientes de sus oraciones. Así es posible medir la similitud entre el resumen S y el centro de la colección y entre el resumen S y cada oración $s_i \in S$. La Ecuación (7) formula esta función objetivo a maximizar, para la que un valor mayor implica una mayor relevancia.

$$\Phi_{Re}(X) = sim(S^*, O) \cdot \left(\sum_{i=1}^{n} sim(S^*, s_i) \cdot x_i \right). \tag{7}$$

III-D. Coherencia

La cuarta función objetivo es la coherencia. Según ella, las oraciones del resumen deben tener una conectividad fluida entre ellas con el fin de mejorar su lectura y comprensión [9]. Esta fluidez se consigue mejorando la similitud entre las oraciones adyacentes en el resumen. Esto se formula como la similitud entre cada una de las oraciones adyacentes s_i y s_{i+1} del resumen en su forma ordenada S', dividida por el número de pares de oraciones comparadas. La forma ordenada del resumen se obtiene mediante un proceso de ordenación de oraciones. El algoritmo de ordenación empleado aquí es el

algoritmo tipo B descrito en [17]. Finalmente, la Ecuación (8) presenta la función objetivo a maximizar.

$$\Phi_{Co}(X) = \frac{\sum_{i=1}^{n_S-1} sim(s_i, s_{i+1})}{n_S - 1}. \quad (8)$$

III-E. Formulación del problema

En el campo de la optimización multi-objetivo, un problema de optimización contiene dos o más funciones objetivo a ser optimizadas simultáneamente. En este caso, se han considerado todas las combinaciones posibles de las cuatro funciones objetivo. Además, la longitud prefijada L (con una cierta tolerancia ε) es una restricción a tener en cuenta. El objetivo principal del problema de resumen extractivo multi-documento es generar un resumen S a partir de la colección D. La formulación general para todas las posibles combinaciones de funciones objetivo es la siguiente:

$$\text{máx } \Phi(X) = \{\Phi_1(X), \ldots, \Phi_f(X)\}, \quad (9)$$

$$\text{sujeto a } L - \varepsilon \leq \sum_{i=1}^{n} l_i \cdot x_i \leq L + \varepsilon, \quad (10)$$

siendo f = 2, 3 o 4, dependiendo de si se consideran 2, 3 o 4 funciones objetivo, y l_i la longitud de la oración s_i. La tolerancia ε se calcula como la diferencia de longitud entre la oración más larga y la más corta.

IV. Metodología

IV-A. Conjuntos de datos

Para los experimentos se ha utilizado los conjuntos de datos de DUC [10], que es un banco de pruebas para la evaluación de resúmenes automáticos. Concretamente, se han usado 10 temas del conjunto de datos DUC2002 (del d061j al d070f), y la longitud del resumen establecida es de 200 palabras. Los documentos de entrada fueron preprocesados mediante los siguientes pasos: segmentación en oraciones, división en palabras, eliminación de las palabras vacías y stemming de las palabras.

IV-B. Métricas de evaluación

Para medir la calidad de los resúmenes generados se han empleado las métricas ROUGE [11], consideradas como oficiales por DUC. En particular, se han utilizado ROUGE-2 y ROUGE-L.

IV-C. Métodos estadísticos

Además de la media de las puntuaciones ROUGE (M), se han utilizado otras dos métricas estadísticas. El rango (R) es la diferencia entre la mejor y la peor puntuación ROUGE, e informa sobre la dispersión de los datos. Sin embargo, este valor no es adimensional. Como medida de dispersión adimensional se ha utilizado una modificación del coeficiente de variación de Pearson. Este coeficiente de variación (CV) se expresa como el cociente (multiplicado por 100) entre el rango y la media de las puntuaciones ROUGE. Además, se ha llevado a cabo un análisis estadístico de las puntuaciones ROUGE mediante pruebas de hipótesis. Se han aplicado pruebas de hipótesis estadísticas para comprobar si existen diferencias estadísticamente significativas. Como las condiciones de normalidad y homocedasticidad del ANOVA uni-factorial no se pueden asumir para ningún tema, se ha empleado el test de Kruskal-Wallis. Cuando los resultados son significativos, las comparaciones múltiples por pares se realizan mediante la aplicación del test de Conover con la corrección de Bonferroni. Se han considerado como estadísticamente significativas aquellas pruebas donde los p-valores son menores que 0,05.

IV-D. Algoritmo usado en los experimentos

El algoritmo MOABC fue desarrollado para este problema en [6]. Este algoritmo se ha adaptado para la optimización de todas las combinaciones de funciones objetivo. Sus parámetros son los siguientes: probabilidad de mutación, $pro_m = 0,1$; número de intentos para mejorar una solución, $lim_n = 10$; tamaño de población, $pob_t = 64$; y número de ciclos, $cic_n = 1000$.

V. Resultados experimentales

V-A. Configuración de los experimentos

La Tabla I contiene las 11 posibles combinaciones de funciones objetivo con los cuatro criterios estudiados, incluyendo también la notación usada para cada una de ellas. Además, el número de repeticiones de cada experimento es $rep_n = 20$.

Tabla I
Composición de las combinaciones de funciones objetivo.

Conjunto de funciones objetivo	Función objetivo				Notación
	CC	RR	Re	Co	
$\Phi(X) = \{\Phi_{CC}(X), \Phi_{RR}(X)\}$	✓	✓			Φ_{1100}
$\Phi(X) = \{\Phi_{CC}(X), \Phi_{Re}(X)\}$	✓		✓		Φ_{1010}
$\Phi(X) = \{\Phi_{CC}(X), \Phi_{Co}(X)\}$	✓			✓	Φ_{1001}
$\Phi(X) = \{\Phi_{RR}(X), \Phi_{Re}(X)\}$		✓	✓		Φ_{0110}
$\Phi(X) = \{\Phi_{RR}(X), \Phi_{Co}(X)\}$		✓		✓	Φ_{0101}
$\Phi(X) = \{\Phi_{Re}(X), \Phi_{Co}(X)\}$			✓	✓	Φ_{0011}
$\Phi(X) = \{\Phi_{CC}(X), \Phi_{RR}(X), \Phi_{Re}(X)\}$	✓	✓	✓		Φ_{1110}
$\Phi(X) = \{\Phi_{CC}(X), \Phi_{RR}(X), \Phi_{Co}(X)\}$	✓	✓		✓	Φ_{1101}
$\Phi(X) = \{\Phi_{CC}(X), \Phi_{Re}(X), \Phi_{Co}(X)\}$	✓		✓	✓	Φ_{1011}
$\Phi(X) = \{\Phi_{RR}(X), \Phi_{Re}(X), \Phi_{Co}(X)\}$		✓	✓	✓	Φ_{0111}
$\Phi(X) = \{\Phi_{CC}(X), \Phi_{RR}(X), \Phi_{Re}(X), \Phi_{Co}(X)\}$	✓	✓	✓	✓	Φ_{1111}

V-B. Combinaciones de dos funciones objetivo

La Tabla II muestra la media, el rango y el CV del ROUGE-2 y ROUGE-L para cada tema de las 6 combinaciones de dos funciones objetivo. En ella se aprecia que Φ_{1100} y Φ_{0110} consiguen los mejores valores en 6 temas para ambos ROUGE, siendo Φ_{1100} la que consigue los mejores valores promedio. En conclusión, cuando solo se optimizan dos funciones objetivo, la reducción de la redundancia es indispensable, mientras que la otra función puede ser la cobertura del contenido o la relevancia, siendo la cobertura del contenido algo mejor.

En cuanto al análisis estadístico de los resultados, las comparaciones por pares se han realizado entre Φ_{1100} y el resto. Los resultados de este análisis determinan que Φ_{1100} obtiene diferencias estadísticamente significativas con el resto en al menos 8 temas tanto en ROUGE-2 como en ROUGE-L.

Tabla II
MEDIA, RANGO Y CV DE ROUGE-2 Y ROUGE-L PARA LAS COMBINACIONES DE DOS FUNCIONES OBJETIVO (EN NEGRITA LOS MEJORES VALORES).

Tema	Φ_{1100}						Φ_{1010}						Φ_{1001}					
	ROUGE-2			ROUGE-L			ROUGE-2			ROUGE-L			ROUGE-2			ROUGE-L		
	M	R	CV	M	M	CV	M	R	CV	M	R	CV	M	R	CV	M	R	CV
d061j	**0,534**	0,000	0,00	**0,692**	0,000	0,00	0,257	0,012	4,74	0,541	0,005	0,90	0,254	0,000	0,00	0,541	0,000	0,00
d062j	**0,436**	0,000	0,00	**0,598**	0,000	0,00	0,329	0,010	3,04	0,518	0,010	1,93	0,333	0,000	0,00	0,526	0,020	3,80
d063j	**0,307**	0,000	0,00	**0,521**	0,000	0,00	0,182	0,030	16,41	0,427	0,032	7,56	0,284	0,000	0,00	0,516	0,000	0,00
d064j	0,344	0,000	0,00	0,534	0,000	0,00	0,200	0,113	56,70	0,399	0,095	23,93	0,261	0,111	42,64	0,443	0,087	19,61
d065j	0,200	0,000	0,00	0,469	0,000	0,00	0,144	0,048	33,62	0,391	0,053	13,47	0,150	0,040	26,51	0,423	0,000	0,00
d066j	**0,318**	0,000	0,00	**0,533**	0,000	0,00	0,265	0,071	26,79	0,518	0,059	11,39	0,256	0,000	0,00	0,509	0,002	0,46
d067f	0,335	0,000	0,00	0,536	0,000	0,00	0,190	0,005	2,62	0,429	0,012	2,88	0,186	0,000	0,00	0,430	0,000	0,00
d068f	**0,391**	0,000	0,00	**0,587**	0,000	0,00	0,276	0,022	8,01	0,471	0,076	16,10	0,297	0,000	0,00	0,513	0,000	0,00
d069f	0,240	0,000	0,00	0,553	0,000	0,00	0,244	0,023	9,32	0,566	0,028	4,88	0,232	0,008	3,27	0,534	0,015	2,82
d070f	**0,251**	0,000	0,00	**0,491**	0,000	0,00	0,248	0,012	4,70	0,475	0,007	1,46	0,249	0,000	0,00	0,477	0,000	0,00
Media	**0,336**	0,000	0,00	**0,551**	0,000	0,00	0,232	0,032	15,66	0,469	0,035	8,01	0,250	0,016	7,24	0,494	0,012	2,67

Tema	Φ_{0110}						Φ_{0101}						Φ_{0011}					
	ROUGE-2			ROUGE-L			ROUGE-2			ROUGE-L			ROUGE-2			ROUGE-L		
	M	R	CV	M	M	CV	M	R	CV	M	R	CV	M	R	CV	M	R	CV
d061j	0,480	0,078	16,25	0,646	0,051	7,90	0,434	0,222	51,17	0,631	0,182	28,85	0,266	0,000	0,00	0,541	0,000	0,00
d062j	0,386	0,000	0,00	0,574	0,000	0,00	0,290	0,090	31,09	0,500	0,082	16,46	0,328	0,013	3,82	0,529	0,030	5,65
d063j	**0,307**	0,000	0,00	**0,521**	0,000	0,00	0,211	0,035	16,58	0,454	0,035	7,64	0,284	0,000	0,00	0,516	0,000	0,00
d064j	**0,355**	0,000	0,00	**0,557**	0,000	0,00	0,331	0,076	23,03	0,531	0,061	11,44	0,255	0,111	43,57	0,438	0,087	19,80
d065j	**0,242**	0,000	0,00	**0,480**	0,004	0,91	0,196	0,051	25,83	0,440	0,035	7,98	0,187	0,000	0,00	0,447	0,015	3,44
d066j	0,220	0,000	0,00	0,488	0,000	0,00	0,306	0,000	0,00	0,513	0,015	2,76	0,256	0,000	0,00	0,509	0,002	0,46
d067f	**0,360**	0,000	0,00	**0,583**	0,000	0,00	0,236	0,134	56,67	0,496	0,136	27,37	0,275	0,000	0,00	0,536	0,000	0,00
d068f	0,248	0,000	0,00	0,543	0,000	0,00	0,338	0,106	31,30	0,557	0,073	13,17	0,280	0,030	10,55	0,474	0,044	9,29
d069f	**0,306**	0,008	2,48	**0,574**	0,003	0,44	0,195	0,051	25,96	0,489	0,070	14,38	0,234	0,013	5,40	0,536	0,040	7,50
d070f	**0,251**	0,000	0,00	**0,491**	0,000	0,00	0,250	0,030	12,02	0,489	0,007	1,42	0,249	0,000	0,00	0,477	0,000	0,00
Media	0,316	0,009	1,87	0,546	0,006	0,92	0,279	0,079	27,36	0,510	0,070	13,15	0,261	0,017	6,33	0,500	0,022	4,61

Tabla III
MEDIA, RANGO Y CV DE ROUGE-2 Y ROUGE-L PARA LAS COMBINACIONES DE TRES FUNCIONES OBJETIVO (EN NEGRITA LOS MEJORES VALORES).

Tema	Φ_{1110}						Φ_{1101}						Φ_{1011}						Φ_{0111}					
	ROUGE-2			ROUGE-L			ROUGE-2			ROUGE-L			ROUGE-2			ROUGE-L			ROUGE-2			ROUGE-L		
	M	R	CV	M	R	CV	M	R	CV	M	R	CV	M	R	CV	M	R	CV	M	R	CV	M	R	CV
d061j	0,536	0,012	2,27	0,692	0,000	0,00	**0,540**	0,007	1,35	**0,714**	0,000	0,00	0,266	0,000	0,00	0,541	0,000	0,00	0,506	0,068	13,49	0,679	0,019	2,86
d062j	**0,436**	0,000	0,00	**0,616**	0,000	0,00	0,431	0,000	0,00	**0,616**	0,000	0,00	0,333	0,000	0,00	0,530	0,022	4,24	0,426	0,000	0,00	**0,616**	0,000	0,00
d063j	0,279	0,000	0,00	0,516	0,000	0,00	0,284	0,000	0,00	0,516	0,000	0,00	0,307	0,012	4,06	0,516	0,040	7,69	**0,317**	0,000	0,00	**0,521**	0,000	0,00
d064j	**0,355**	0,000	0,00	**0,557**	0,000	0,00	0,353	0,000	0,00	**0,557**	0,000	0,00	0,255	0,111	43,57	0,438	0,087	19,80	0,353	0,000	0,00	**0,557**	0,000	0,00
d065j	0,207	0,000	0,00	0,469	0,000	0,00	**0,216**	0,000	0,00	0,469	0,000	0,00	**0,216**	0,000	0,00	**0,490**	0,004	0,90	0,200	0,002	1,10	0,444	0,002	0,50
d066j	**0,325**	0,000	0,00	**0,552**	0,000	0,00	0,315	0,000	0,00	0,533	0,000	0,00	0,256	0,000	0,00	0,509	0,002	0,46	0,320	0,000	0,00	**0,552**	0,000	0,00
d067f	**0,360**	0,000	0,00	**0,583**	0,000	0,00	0,337	0,000	0,00	0,536	0,000	0,00	0,275	0,000	0,00	0,536	0,000	0,00	0,359	0,022	6,23	0,580	0,047	8,08
d068f	0,400	0,000	0,00	**0,638**	0,000	0,00	0,399	0,007	1,85	0,615	0,007	1,19	0,297	0,000	0,00	0,543	0,000	0,00	**0,403**	0,000	0,00	**0,638**	0,000	0,00
d069f	0,240	0,000	0,00	0,553	0,000	0,00	0,240	0,000	0,00	0,553	0,000	0,00	**0,245**	0,000	0,00	**0,568**	0,003	0,44	0,239	0,003	1,06	0,553	0,000	0,00
d070f	0,251	0,000	0,00	0,491	0,000	0,00	**0,267**	0,000	0,00	**0,502**	0,000	0,00	0,249	0,000	0,00	0,477	0,000	0,00	**0,267**	0,000	0,00	**0,502**	0,000	0,00
Media	**0,339**	0,001	0,23	**0,567**	0,000	0,00	0,338	0,001	0,32	0,561	0,001	0,12	0,270	0,012	4,76	0,515	0,016	3,35	**0,339**	0,010	2,19	0,564	0,007	1,14

V-C. Combinaciones de tres funciones objetivo

Los resultados de las 4 posibles combinaciones de tres funciones objetivo se presentan en la Tabla III. De ella se concluye que Φ_{1110} y Φ_{0111} logran el mejor ROUGE-2 promedio y Φ_{1110} consigue el mejor ROUGE-L promedio. Además, Φ_{1110} consigue el mejor valor en 4 temas para el ROUGE-2, mejorando a las demás. Y en cuanto al ROUGE-L, Φ_{0111} consigue el mejor valor en 6 temas, uno más que Φ_{1110}.

Se han realizado comparaciones por pares de Φ_{1110} con respecto al resto. Para el ROUGE-L, Φ_{1110} obtiene diferencias estadísticamente significativas en los 10 temas con respecto a Φ_{1011}, en 5 con respecto a Φ_{1101} y en 4 con respecto a Φ_{0111}. Analizando las diferencias tema a tema, comparando Φ_{1110} y Φ_{1101}, en 3 de las 5 diferencias la combinación Φ_{1110} consigue mejores resultados. De la misma forma, comparando Φ_{1110} y Φ_{0111}, en 2 de las 4 diferencias Φ_{1110} es mejor. Por lo tanto, Φ_{1110} y Φ_{0111} obtienen resultados muy similares para ROUGE-L. En cuanto a los resultados del ROUGE-2, los de Φ_{1110} son mejores, con diferencias estadísticamente significativas en al menos 8 de los temas, por lo que es la mejor combinación de tres funciones objetivo. Estos resultados apoyan las conclusiones de la subsección anterior, en la que los objetivos de la reducción de la redundancia, la cobertura del contenido y la relevancia (en este orden) fueron los más importantes.

V-D. Combinaciones de cuatro funciones objetivo

Los resultados obtenidos para la única combinación de cuatro funciones objetivo se presentan en la Tabla IV.

V-E. Comparación de las mejores combinaciones

Los resultados mostrados en las Tablas II, III y IV concluyen que Φ_{1111} es la que obtiene los mejores valores promedio en ambos ROUGE. Consigue el mejor ROUGE-2 en 6 temas, y el mejor ROUGE-L en 8 temas. Además, Φ_{1110} consigue el segundo mejor valor promedio en ambos ROUGE.

También se han realizado comparaciones por pares. Los resultados de este análisis informan que Φ_{1100} y Φ_{1110} obtienen diferencias estadísticamente significativas en 6 temas (para ambos ROUGE) y que Φ_{1100} y Φ_{1111} obtienen diferencias

Tabla IV
MEDIA, RANGO Y CV DE ROUGE-2 Y ROUGE-L PARA LA COMBINACIÓN DE CUATRO FUNCIONES OBJETIVO.

Tema	Φ_{1111}					
	ROUGE-2			ROUGE-L		
	M	R	CV	M	R	CV
d061j	0,552	0,041	7,51	0,725	0,056	7,70
d062j	0,431	0,000	0,00	0,616	0,000	0,00
d063j	0,284	0,000	0,00	0,516	0,000	0,00
d064j	0,353	0,000	0,00	0,557	0,000	0,00
d065j	0,215	0,011	5,13	0,467	0,022	4,69
d066j	0,320	0,000	0,00	0,552	0,000	0,00
d067f	0,360	0,000	0,00	0,583	0,000	0,00
d068f	0,403	0,000	0,00	0,638	0,000	0,00
d069f	0,240	0,003	1,06	0,553	0,000	0,00
d070f	0,267	0,000	0,00	0,502	0,000	0,00
Media	0,343	0,005	1,37	0,571	0,008	1,24

significativas en al menos 8 temas (para ambos ROUGE). En cuanto a la comparación entre Φ_{1110} y Φ_{1111}, se producen diferencias en 8 temas para ROUGE-2 y en 2 temas para ROUGE-L. En este último caso se han analizado las diferencias tema a tema, siendo Φ_{1111} mejor en los 2 temas. Por lo tanto, Φ_{1111} es la que ha logrado los mejores resultados promedio, obteniendo además diferencias estadísticamente significativas con las otras dos combinaciones.

A continuación se analizan las mejoras producidas por cada combinación. Φ_{1110} mejora a Φ_{1100}, al igual que Φ_{1111} a Φ_{1110}. Esto es, el resultado mejora a medida que lo hace el número de funciones objetivo optimizadas, aunque estas mejoras son pequeñas. Por este motivo, se ha analizado el tiempo de ejecución de estas tres combinaciones con el fin de estudiar el posible incremento del coste computacional. La Tabla V muestra los tiempos de ejecución en segundos para cada tema, además del tiempo medio de cada combinación. En ella se puede apreciar que Φ_{1100} es la que obtiene el menor tiempo de ejecución, seguida muy de cerca por Φ_{1110}. Por otro lado, Φ_{1111} obtiene un tiempo de ejecución mucho mayor.

Tabla V
TIEMPOS DE EJECUCIÓN MEDIOS (S) DE LAS MEJORES COMBINACIONES.

Tema	Φ_{1100}	Φ_{1110}	Φ_{1111}
d061j	3.984,09	4.122,20	11.578,99
d062j	1.426,87	1.536,60	4.448,33
d063j	9.365,52	9.592,35	28.049,58
d064j	5.378,22	5.732,15	15.943,38
d065j	15.313,65	16.001,39	47.726,48
d066j	5.685,67	5.999,84	16.858,20
d067f	1.503,26	1.586,80	4.686,46
d068f	1.517,97	1.554,74	4.591,76
d069f	24.898,45	25.753,85	76.837,68
d070f	2.292,36	2.395,31	6.795,54
Media	7.136,61	7.427,52	21.751,64

Ahora, con el fin de analizar el tiempo de ejecución en relación al número de funciones objetivo a optimizar, se compara Φ_{1100} con respecto a Φ_{1110} y Φ_{1111}. La razón de elegir a Φ_{1100} como referencia es porque es la más utilizada en la literatura científica. La Tabla VI presenta la comparación global entre ellas, mostrando en porcentaje las mejoras producidas en las puntuaciones ROUGE y el aumento del tiempo de ejecución. De ella se concluye que Φ_{1110}, que incluye el criterio de la relevancia, mejora los resultados del ROUGE-2 y ROUGE-L en 0,89 % y en 2,90 % respectivamente, mientras que el tiempo de ejecución se incrementa solo un 4,08 %. En cuanto a Φ_{1111}, que contiene además el criterio de la coherencia, mejora el ROUGE-2 en un 2,08 % y el ROUGE-L en un 3,63 %. Sin embargo, el tiempo de ejecución aumenta en un 204,79 %, esto es, es tres veces más lenta. Este incremento se debe al coste de optimizar la función objetivo de la coherencia, la cual conlleva la ejecución del algoritmo de ordenación de oraciones en cada cálculo. Para concluir qué combinación es la mejor, es necesario relacionar la mejora de los resultados ROUGE con el aumento del tiempo de ejecución. Para Φ_{1110}, una mejora del 1 % en ROUGE-2 y en ROUGE-L con respecto a Φ_{1100} implica un aumento del tiempo de ejecución del 4,58 % y del 1,41 % respectivamente. Sin embargo, para Φ_{1111}, una mejora del 1 % implica aumentar un 98,46 % y un 56,42 % respectivamente el tiempo de ejecución. Por lo tanto, estos ratios revelan que Φ_{1110} es la más balanceada en términos de ROUGE promedio y tiempo de ejecución.

Tabla VI
COMPARACIÓN GLOBAL DE LAS COMBINACIONES Φ_{1110} Y Φ_{1111} CON RESPECTO A LA COMBINACIÓN Φ_{1100}.

Combinación	ROUGE-2	ROUGE-L	Tiempo de ejecución
Φ_{1110}	+0,89 %	+2,90 %	+4,08 %
Φ_{1111}	+2,08 %	+3,63 %	+204,79 %

V-F. Comparación con otros trabajos

Por último, se presenta una comparación entre los resultados de la mejor combinación y los de otros trabajos. Los resultados obtenidos por la combinación Φ_{1110} se han comparado con los resultados obtenidos en [5] y en [7]. El primero fue el mejor enfoque mono-objetivo, usando un algoritmo DE adaptativo, y el segundo fue el mejor enfoque multi-objetivo, empleando el algoritmo NSGA-II. La Tabla VII presenta la comparación entre estos resultados. De ella se concluye que Φ_{1110} consigue los mejores resultados en ambos ROUGE: en ROUGE-2 mejora en 9 temas a [5] y en 8 a [7], y en ROUGE-L mejora en 8 temas a ambos trabajos. Comparando ahora los resultados promedio, Φ_{1110} obtiene una mejora del 42,44 % y del 28,90 % en ROUGE-2 con respecto a [5] y [7] respectivamente. En ROUGE-L las mejoras son del 13,86 % y del 11,39 % respectivamente.

VI. CONCLUSIONES

El problema de resumen extractivo multi-documento se ha formulado tradicionalmente como un problema que optimiza los criterios de la cobertura del contenido y de la reducción de la redundancia. Otros criterios como la relevancia y la coherencia también se han utilizado, pero en menor medida. Se ha llevado a cabo un estudio comparativo de todas las combinaciones de los diferentes criterios para establecer cuáles son los más apropiados para este problema. Se han analizado un total de once combinaciones, cuyos resultados se han tratado estadísticamente. Los resultados obtenidos han demostrado que la reducción de la redundancia es la función objetivo más influyente, ya que las combinaciones en las que está presente

Tabla VII
COMPARACIÓN DE LA MEDIA, RANGO Y CV DE ROUGE-2 Y ROUGE-L ENTRE LA COMBINACIÓN Φ_{1110} Y LOS OTROS TRABAJOS (LOS MEJORES VALORES EN NEGRITA).

Tema	ROUGE-2									ROUGE-L								
	DE adaptativo [5]			NSGA-II [7]			Combinación Φ_{1110}			DE adaptativo [5]			NSGA-II [7]			Φ_{1110}		
	M	R	CV	M	R	CV	M	R	CV	M	R	CV	M	R	CV	M	R	CV
d061j	0,266	0,290	109,02	0,306	0,263	85,95	**0,536**	0,012	2,27	0,542	0,208	38,38	0,554	0,205	37,00	**0,692**	0,000	0,00
d062j	0,188	0,275	146,28	0,200	0,422	211,00	**0,436**	0,000	0,00	0,473	0,239	50,53	0,481	0,306	63,62	**0,616**	0,000	0,00
d063j	0,245	0,208	84,90	0,275	0,279	101,45	**0,279**	0,000	0,00	0,493	0,156	31,64	**0,528**	0,171	32,39	0,516	0,000	0,00
d064j	0,194	0,280	144,33	0,233	0,356	152,79	**0,355**	0,000	0,00	0,462	0,235	50,87	0,488	0,287	58,81	**0,557**	0,000	0,00
d065j	0,144	0,209	145,14	0,182	0,208	114,29	**0,207**	0,000	0,00	0,431	0,141	32,71	0,457	0,174	38,07	**0,469**	0,000	0,00
d066j	0,201	0,257	127,86	0,181	0,245	135,36	**0,325**	0,000	0,00	0,455	0,196	43,08	0,441	0,149	33,79	**0,552**	0,000	0,00
d067f	0,239	0,235	98,33	0,260	0,298	114,62	**0,360**	0,000	0,00	0,509	0,232	45,58	0,529	0,244	46,12	**0,583**	0,000	0,00
d068f	0,491	0,384	78,21	**0,496**	0,281	56,65	0,400	0,000	0,00	**0,666**	0,226	33,93	0,626	0,226	36,10	0,638	0,000	0,00
d069f	0,184	0,166	90,22	0,232	0,239	103,02	**0,240**	0,000	0,00	0,454	0,135	29,74	0,476	0,191	40,13	**0,553**	0,000	0,00
d070f	0,224	0,260	116,07	**0,262**	0,215	82,03	0,251	0,000	0,00	0,496	0,173	34,88	**0,513**	0,158	30,80	0,491	0,000	0,00
Media	0,238	0,256	114,03	0,263	0,281	115,72	**0,339**	0,001	0,23	0,498	0,194	39,13	0,509	0,211	41,68	**0,567**	0,000	0,00

obtienen las mejores puntuaciones ROUGE-2 y ROUGE-L. Por otro lado, la coherencia es la función objetivo que menos influye en el resultado final. En cuanto a las funciones objetivo de la cobertura del contenido y de la relevancia, ambas proporcionan buenos resultados. Sin embargo, los de la cobertura del contenido son ligeramente mejores. Además, los resultados mejoran cuando el número de funciones objetivo a optimizar es mayor. La inclusión de las funciones objetivo de la relevancia y de la coherencia, junto con las de la cobertura del contenido y la de la reducción de la redundancia, mejora los resultados aunque aumenta considerablemente el tiempo de ejecución. En conclusión, la combinación formada por la cobertura del contenido, la reducción de la redundancia y la relevancia ha obtenido los resultados más equilibrados en términos de rendimiento ROUGE y tiempo de ejecución.

Como trabajo futuro, esta combinación se implementará en el software NeuroK [18], que es una plataforma de e-learning basada en los principios de la neurodidáctica y de las redes sociales, con el fin de generar resúmenes automáticamente a partir del contenido textual escrito por los alumnos en un curso virtual. Esto permitirá a los profesores supervisar más fácilmente el proceso de aprendizaje de los alumnos en la plataforma.

AGRADECIMIENTOS

Esta investigación ha sido parcialmente financiada por el Ministerio de Ciencia, Innovación y Universidades y la Agencia Estatal de Investigación (proyectos PID2019-107299GB-I00/AEI/10.13039/501100011033 y MTM2017-86875-C3-2-R), por la Junta de Extremadura (proyectos GR18090 y GR18108) y por la Unión Europea (Fondo Europeo de Desarrollo Regional). Jesus M. Sanchez-Gomez dispone de un Contrato Predoctoral financiado por la Junta de Extremadura (contrato PD18057) y la Unión Europea (Fondo Social Europeo).

REFERENCIAS

[1] W. Fan and A. Bifet, "Mining big data: current status, and forecast to the future," *ACM SIGKDD Explorations Newsletter*, vol. 14, no. 2, pp. 1–5, 2013.

[2] H. Hashimi, A. Hafez, and H. Mathkour, "Selection criteria for text mining approaches," *Comput. Hum. Behav.*, vol. 51, pp. 729–733, 2015.

[3] D. Zajic, B. Dorr, and J. Lin, "Single-document and multi-document summarization techniques for email threads using sentence compression," *Inf. Process. Manage.*, vol. 44, no. 4, pp. 1600–1610, 2008.

[4] X. Wan, "An exploration of document impact on graph-based multi-document summarization," in *Proceedings of the Conference on Empirical Methods in NLP*. ACL, 2008, pp. 755–762.

[5] R. Alguliev, R. Aliguliyev, and C. Mehdiyev, "Sentence selection for generic document summarization using an adaptive differential evolution algorithm," *Swarm Evol. Comput.*, vol. 1, no. 4, pp. 213–222, 2011.

[6] J. M. Sanchez-Gomez, M. A. Vega-Rodríguez, and C. J. Pérez, "Extractive multi-document text summarization using a multi-objective artificial bee colony optimization approach," *Knowledge-Based Syst.*, vol. 159, pp. 1–8, 2018.

[7] H. H. Saleh, N. J. Kadhim, and B. A. Attea, "A Genetic Based Optimization Model for Extractive Multi-Document Text Summarization," *Iraqi Journal of Science*, vol. 56, no. 2, pp. 1489–1498, 2015.

[8] R. Alguliev, R. Aliguliyev, and N. Isazade, "DESAMC+DocSum: Differential evolution with self-adaptive mutation and crossover parameters for multi-document summarization," *Knowledge-Based Syst.*, vol. 36, pp. 21–38, 2012.

[9] K. Umam, F. W. Putro, G. Q. O. Pratamasunu, A. Z. Arifin, and D. Purwitasari, "Coverage, Diversity, and Coherence Optimization for Multi-Document Summarization," *Jurnal Ilmu Komputer dan Informasi*, vol. 8, no. 1, pp. 1–10, 2015.

[10] "Document Understanding Conferences," http://duc.nist.gov, 2014, last accessed: 17-June-2021.

[11] C.-Y. Lin, "ROUGE: A package for automatic evaluation of summaries," in *Text summarization branches out: Proceedings of the ACL-04 workshop*, vol. 8. Barcelona, Spain, 2004, pp. 74–81.

[12] R. Alguliev, R. Aliguliyev, and M. Hajirahimova, "GenDocSum+MCLR: Generic document summarization based on maximum coverage and less redundancy," *Expert Syst. Appl.*, vol. 39, no. 16, pp. 12 460–12 473, 2012.

[13] R. Alguliev, R. Aliguliyev, and N. Isazade, "An unsupervised approach to generating generic summaries of documents," *Appl. Soft. Comput.*, vol. 34, pp. 236–250, 2015.

[14] R. Alguliyev, R. Aliguliyev, N. Isazade, A. Abdi, and N. Idris, "COSUM: Text summarization based on clustering and optimization," *Expert Systems*, vol. 36, no. 1, p. e12340, 2019.

[15] P. Verma and H. Om, "MCRMR: Maximum coverage and relevancy with minimal redundancy based multi-document summarization," *Expert Syst. Appl.*, vol. 120, pp. 43–56, 2019.

[16] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Inf. Process. Manage.*, vol. 24, no. 5, pp. 513–523, 1988.

[17] D. Bollegala, N. Okazaki, and M. Ishizuka, "A preference learning approach to sentence ordering for multi-document summarization," *Inf. Sci.*, vol. 217, pp. 78–95, 2012.

[18] F. Calle-Alonso, A. Cuenca-Guevara, D. de la Mata Lara, J. M. Sanchez-Gomez, M. A. Vega-Rodríguez, and C. J. Perez Sanchez, "NeuroK: A Collaborative e-Learning Platform based on Pedagogical Principles from Neuroscience," in *Proceedings of the 9th International Conference on CSEDU*, vol. 1. SciTePress, 2017, pp. 550–555.

Algoritmos Evolutivos con Memoria Elite para la Selección de Carteras de Inversión

Feijoo Colomine Durán
Universidad Nacional Experimental del Táchira (UNET),
Laboratorio de Computación de Alto Rendimiento (LCAR),
San Cristóbal, Venezuela
Email: fcolomin@unet.edu.ve

Carlos Cotta*† and Antonio J. Fernández-Leiva*†
*ITIS Software, Universidad de Málaga
†Dept. Lenguajes y Ciencias de la Computación,
ETSI Informática, Universidad de Málaga,
Campus de Teatinos, 29071 - Málaga, España.
Email: {ccottap,afdez}@lcc.uma.es

***Abstract*—La presente investigación considera la implementación de un algoritmo evolutivo (EA) con variación poblacional basado en memoria élite (ME) para resolver la selección de carteras de inversión con restricciones de cardinalidad. Se trata de un problema de optimización intrínsecamente multiobjetivo para el que utilizamos el índice de Sharpe, una conocida medida de rentabilidad-riesgo, para evaluar la eficiencia financiera de las carteras de inversión generadas como mecanismo de solución, así como para realizar análisis comparativos. El índice de Sharpe también está en el núcleo de la memoria de élite utilizada por el EA, para mantener un registro de las soluciones más efectivas encontradas. Este registro se utiliza como un mecanismo de intensificación sintonizable para mantener la búsqueda enfocada en regiones de interés. Se ha realizado un análisis de sensibilidad del algoritmo para determinar una parametrización adecuada. Se ha llevado a cabo asimismo una comparación con tres algoritmos evolutivos multiobjetivo utilizando indicadores de desempeño bien conocidos como el hipervolumen y la distancia generacional. Esta experimentación toma como referencia el mercado financiero accionario de la Bolsa de Valores de Colombia entre 2010-2016, y confirma el gran potencial del uso de la ME para este problema.**

***Index Terms*—Algoritmos Evolutivos, Problemas Multiobjetivo, Carteras de Inversión, Memoria Elite.**

I. Introducción

Existen en el área de gestión de instrumentos financieros toda una serie de elementos teóricos y estudios relacionados con la relación riesgo-rendimiento [1]. El potencial de rendimiento o pérdida de las inversiones no es estático, sino que depende siempre de la evolución del mercado. En la literatura se pueden encontrar diversas propuestas que modelan este escenario; por ejemplo, desde su aparición el modelo de Markowitz [2] se ha convertido en una referencia teórica fundamental para la selección de carteras de inversión. Sin embargo su aplicación en la práctica no ha sido tan extensa, debido, principalmente a la complejidad matemática del método.

Cuando a este modelo se le agregan algunas otras restricciones como por ejemplo las de cardinalidad se convierte en un modelo cuya complejidad debe ser abordada con técnicas de búsqueda de soluciones cuasi óptimas. La mayor contribución del presente trabajo es la demostración de que se pueden establecer algunas técnicas de explotación de conocimiento que nos permite enfocarnos en algunas regiones de la frontera de Pareto que se ajustan mejor a lo que se busca para obtener un mejor provecho: maximizar el rendimiento de la cartera de inversión con un bajo riesgo con una técnica basada en el uso de una memoria élite que mantiene un conjunto de soluciones seleccionadas en el contexto de la búsqueda de los mejores individuos en un mecanismo evolutivo.

Esta investigación considera la implementación de un algoritmo evolutivo con cualidades de elitismo e implantación de un método de variación de la población basado en explotación de conocimiento del problema a medida que el algoritmo se va ejecutando. Para esto se ha considerado utilizar una base de conocimiento del problema, esto es, el índice de Sharpe [3]. Este índice es una medida de la relación entre el rendimiento y el riesgo en una cartera de inversión determinada.

Este estudio comprende un análisis de sensibilidad de la mejor parametrización de la propuesta de algoritmo evolutivo y un estudio comparativo con diferentes técnicas algorítmicas para abordar un problema multiobjetivo usando computación evolutiva. Más específicamente el estudio comparativo considera tres algoritmos que son referencia en esta materia: el *Nondominated Sorting Genetic Algorithm II* (NSGA-II) [4], el *Strength Pareto Evolutionary Algorithm 2* (SPEA2) [5] y el *Indicator-Based Selection in Multiobjective Search* (IBEA) [6].

II. Materiales y Métodos

De acuerdo a lo mencionado en la sección I, se ha considerado el uso de diferentes técnicas metaheurísticas para optimización de carteras de inversión con restricciones de cardinalidad en el contexto del referente financiero basado en el modelo de Markowitz. En las siguientes secciones se describirá con mayor detalle qué técnicas se utilizaron para la comparación con esta propuesta que desarrolla su mecanismo de actuación con una medida de rendimiento soportada en el índice de Sharpe. Finalmente, se describirán en detalle la base de datos usada en la experimentación.

A. El problema sujeto de la optimización

El modelo desarrollado por Markowitz [2] tiene como objetivo obtener un conjunto de n activos que conformen una cartera para lograr el mayor rendimiento posible con el menor riesgo. Para esto se estiman las rentabilidades promedio

y su respectiva variabilidad. Además de esto se calculan las diferentes correlaciones entre los activos sujetos de estudio.

El modelo de Markowitz expresado con múltiples objetivos quedaría así:

$$\min \sigma^2(R_p) = \sum_{i=1}^{n}\sum_{j=1}^{n} w_i w_j \sigma_{ij}$$

$$\max E(R_p) = \sum_{i=1}^{n} w_i E(R_i) \quad (1)$$

o

$$\max E(R_p) = \min(-E(R_p)) \quad (2)$$

sujeto a:

$$\sum_{i=1}^{n} w_i = 1$$

donde $w_i \geqslant 0$ es la proporción del presupuesto del inversor destinado al activo financiero i e incógnita del problema, $\sigma^2(R_p)$ es la varianza de la cartera p, y σ_{ij} es la covarianza entre los rendimientos de los valores i y j, y $E(R_p)$ es la rentabilidad o rendimiento esperado de la cartera p.

A este modelo se le pueden agregar algunas otras restricciones, como por ejemplo:

1) Cardinalidad: Un máximo de K pesos es diferente de cero.
2) Limite en el porcentaje de asignación para un activo.

En este trabajo se desarrollaron los métodos basados en las restricciones del primer tipo, esto es, al modelo presentado en la ecuación (1) se le agrega una restricción más que tiene que ver con la cardinalidad, es decir, un máximo de K pesos es diferente de cero. El conjunto de pares [$E(R_p)$,$\sigma^2(R_p)$] o combinaciones rentabilidad-riesgo de todas las carteras eficientes (esto es, de aquellas carteras en las que no puede aumentarse la rentabilidad sin aumentar el riesgo ni disminuir el riesgo sin que la rentabilidad también disminuya) es denominado "frontera eficiente". Una vez conocida ésta, el inversor, de acuerdo con sus preferencias y el nivel de riesgo que está dispuesto a asumir, elegirá su cartera óptima [7]. Al margen de este criterio de índole más subjetivo para cada inversor, existe en la literatura financiera una medida de desempeño que los expertos en finanzas utilizan para cuantificar y comparar que tan buen resultado da una cartera conformada por un conjunto de activos sobre otra. Esta medida es el índice de Sharpe[3]. En la Ecuación (3) se puede visualizar dicha relación:

$$S(\vec{R}|\vec{W}) = \frac{E(\vec{R}|\vec{W}) - R_0}{\sigma(\vec{R}|\vec{W})} \quad (3)$$

donde R_0 es el rendimiento que se supone libre de riesgo. Así, $E(\vec{R}|\vec{W}) - R_0$ es el exceso de rendimiento susceptible de tener algún riesgo. El índice de Sharpe tiene como divisor la cuantía del riesgo de la cartera expresada en la desviación estándar del rendimiento de la cartera. Este índice es una prima por unidad de riesgo, por lo que si el índice es elevado la cartera puede prometer mejores rendimientos a menor riesgo.

El índice de Sharpe puede además tener una interesante interpretación geométrica. La ecuación descrita en (3) es una linea recta que es tangente a la curva determinada por el conjunto optimo de carteras (Frontera de Pareto) en el plano de (X = riesgo, Y = rendimiento) como resultado de la solución del modelo de Markowitz. Además, dadas dos carteras $\vec{W}_1$ y $\vec{W}_2$, si el último domina (en un sentido de Pareto) al primero –i.e., $E(\vec{R}|\vec{W}_1) \leqslant E(\vec{R}|\vec{W}_2)$ y $\sigma(\vec{R}|\vec{W}_1) \geqslant \sigma(\vec{R}|\vec{W}_2)$, con al menos una de las desigualdades siendo estricta– entonces debe tener un valor más alto de índice de Sharpe (porque $\vec{W}_2$ tendría un numerador más grande y un denominador más pequeño en la Ecuación (3). Esto sugiere que el índice de Sharpe se puede utilizar dentro de un optimizador multiobjetivo para avanzar hacia el frente de Pareto.

B. Un enfoque evolutivo

Como se anticipó en la sección I, una de las cuestiones clave para aplicar con éxito metaheurísticas a una determinada tarea de optimización es adaptar la primera a la última tarea específica[8].

Este es precisamente uno de los principios centrales de los algoritmos con características meméticas (MA) [9]. Desde un punto de vista muy específico, estas técnicas surgen de la combinación de algoritmos de optimización basados en la población (a menudo responsables de proporcionar capacidades de exploración / diversificación) y alguna forma de búsqueda local (a su vez, responsable de la explotación/intensificación de la búsqueda en regiones prometedoras del espacio de búsqueda). Si bien esto no agota las posibilidades dentro del marco de los algoritmos meméticos (ver [10], [11]), constituye un esqueleto básico sobre el cual se puede construir un método de optimización altamente efectivo. Lo que es más importante, estos métodos pueden considerarse como una estrategia complementaria de resolución de problemas, destinada a beneficiarse de las ideas algorítmicas existentes, combinándolas de forma sinérgica. En el presente caso, el componente intensificador no se instancia a través de un procedimiento de búsqueda local, sino mediante un mecanismo de memoria élite. Es por ello que podría considerarse como un enfoque memético dentro de la definición más amplia del término [12], si bien emplearemos preferentemente la denominación "evolutivo" en este trabajo para dejar más clara la diferencia con modelos meméticos tradicionales.

El enfoque considerado utiliza el método IBEA [6] como motor de búsqueda evolutivo, debido a su buen desempeño en este dominio del problema [13]. En cuanto al componente intensificador, su funcionamiento y propósito es como sigue: la memoria de soluciones élite es una lista ordenada de los mejores individuos (según el índice de Sharpe) que genera el algoritmo en un momento dado. Esta lista está inicialmente vacía y tiene un cierto tamaño máximo fijo Θ. Cada vez que se genera una nueva solución, se genera su índice de Sharpe. Si es positivo, entonces se prueba con esta memoria de élite: si hay espacio en él, se inserta; de lo contrario, sustituye a la peor solución (según el índice de Sharpe) de la colección, siempre que la primera sea mejor que la segunda.

Algorithm 1: Algoritmo Evolutivo Basado en Sharpe

Parameters:
μ: Tamaño de Población;
λ: número de descendientes;
Θ: Tamaño de la Memoria Elite;
P_ξ: Probabilidad de corrección elite

```
elite ← ⟨⟩;
for i ← 1 to μ do
    pop_i ← GENERATESOLUTION();
    UPDATEELITE(elite, Θ, pop_i);
end for
ξ̄ ← COMPUTEAVGSHARPE(pop);
for i ← 1 to μ do
    if aleatorio(0,1)< P_ξ then
        CORRECTSOLUTION(elite, pop_i, ξ̄);
    end if
end for
while ¬ TERMINATION() do
    offspring ← SELECTANDREPRODUCE(pop);
    for i ← 1 to λ do
        UPDATEELITE(elite, Θ, offspring_i);
    end for
    ξ̄ ← COMPUTEAVGSHARPE(pop);
    for i ← 1 to λ do
        if aleatorio(0,1)< P_ξ then
            CORRECTSOLUTION(elite, offspring_i, ξ̄);
        end if
    end for
    pop ← REPLACE (pop, offspring);
end while
```

El propósito de esta lista es proporcionar una intensificación adicional de la búsqueda. Para ello, el algoritmo utiliza en cada iteración el índice de Sharpe promedio de esta memoria $\bar{\xi}$: para cada nuevo individuo creado en esa generación se activa un procedimiento de corrección con una probabilidad P_ξ: si el nuevo individuo y su índice de Sharpe está por debajo de $\bar{\xi}$ se sustituye por un individuo aleatorio de la memoria de las soluciones de élite. Al hacerlo, se descartan las soluciones por debajo del promedio y la búsqueda se reorienta en torno a soluciones de élite anteriores. Cabe señalar que este procedimiento está destinado a funcionar en conjunto con el procedimiento de mutación, recombinación y de búsqueda local (en caso de que lo hubiera).

Todo el procedimiento está esbozado en el algoritmo 1. De manera análoga al lamarckismo parcial [14], el empleo de la corrección élite se hace con cierta probabilidad con el objetivo de proporcionar un mecanismo de control sobre la intensificación/diversificación del proceso.

C. Datos usados en el análisis

Este trabajo ha considerado los precios de cierre mensuales en la Bolsa de Valores de Colombia (BVC) de veinte fondos de capital durante los años 2010 a 2016. Por lo general, es conveniente elegir un intervalo de al menos 5 años, para –en un escenario ideal– capturar un ciclo de mercado completo con la correspondiente fluctuación en los precios de las acciones. Por otro lado, no es recomendable elegir un intervalo muy grande (por ejemplo, 10 años o más), ya que el estado real del fondo puede no estar bien representado por información de mercado muy antigua. En este sentido, el conjunto de datos considerado se ajusta cómodamente dentro de estos límites. Los datos se han recopilado del sitio web público de la Bolsa de Valores de Colombia [15].

III. Experimentación

En esta sección se muestra el proceso de experimentación desarrollado según el marco definido en la sección II.

A. Configuración de la experimentación

Se han realizado experimentos con los tres algoritmos, a saber, NSGA-II, SPEA2, IBEA y EA con memoria élite (Ver sección:II-B). Se ha utilizado la biblioteca PISA (una interfaz independiente de plataforma y lenguaje de programación para algoritmos de búsqueda) propuesta por [16] para los algoritmos evolutivos multiobjetivos básicos, así como para el motor evolutivo del EA. En todos los casos, los pesos se codifican usando 10 bits (por lo tanto, resulta en una cadena de bits de longitud $\ell = 200$, ya que hay 20 fondos como se indica en la Sección II-C), la tasa de cruce es $P_x = 0.8$, la tasa de mutación es $P_m = 1/\ell$ y el tamaño de la población es 2ℓ. Con respecto al EA, la memoria de la solución de élite tiene un tamaño $\Theta = 30$, y se han considerado valores de la probabilidad de corrección elite $P_\xi \in \{0.00, 0.05, 0.10, 0.20, 0.30, 0.50, 0.80, 0.90, 1.00\}$. Los algoritmos se ejecutaron para un número máximo de 20000 evaluaciones de aptitud, y se realizaron 30 ejecuciones para cada una de ellas para $K = 14$.

Para evaluar los resultados proporcionados por cada método se ha considerado una doble perspectiva, a saber, la calidad del frente de Pareto obtenido y la calidad de la cartera seleccionada de este frente. Para el último aspecto, se considera el índice de Sharpe como se menciona en la Sección II-A. En cuanto a la evaluación de los frentes de Pareto, se utilizaron dos indicadores de desempeño bien conocidos: el indicador de hipervolumen [17] y el indicador de distancia generacional GD [18]. El primero proporciona una indicación de la región en el espacio de búsqueda que está dominada por el frente (por lo tanto, cuanto más grande, mejor). Esta medida requiere un punto de referencia que en este caso se define por el riesgo máximo y el beneficio mínimo observado en las soluciones en el frente de Pareto más conocido (el frente de Pareto combinado obtenido por todos los algoritmos para el valor particular de K considerado). En cuanto al segundo indicador, estima la medida en que un cierto frente se aproxima a otro (el verdadero frente óptimo de Pareto si se conoce, o un frente de referencia en caso contrario). Se ha considerado la versión unaria de este indicador, tomando nuevamente el frente de Pareto combinado (esto es, el frente de Pareto obtenido mediante la unión de todas las soluciones encontradas

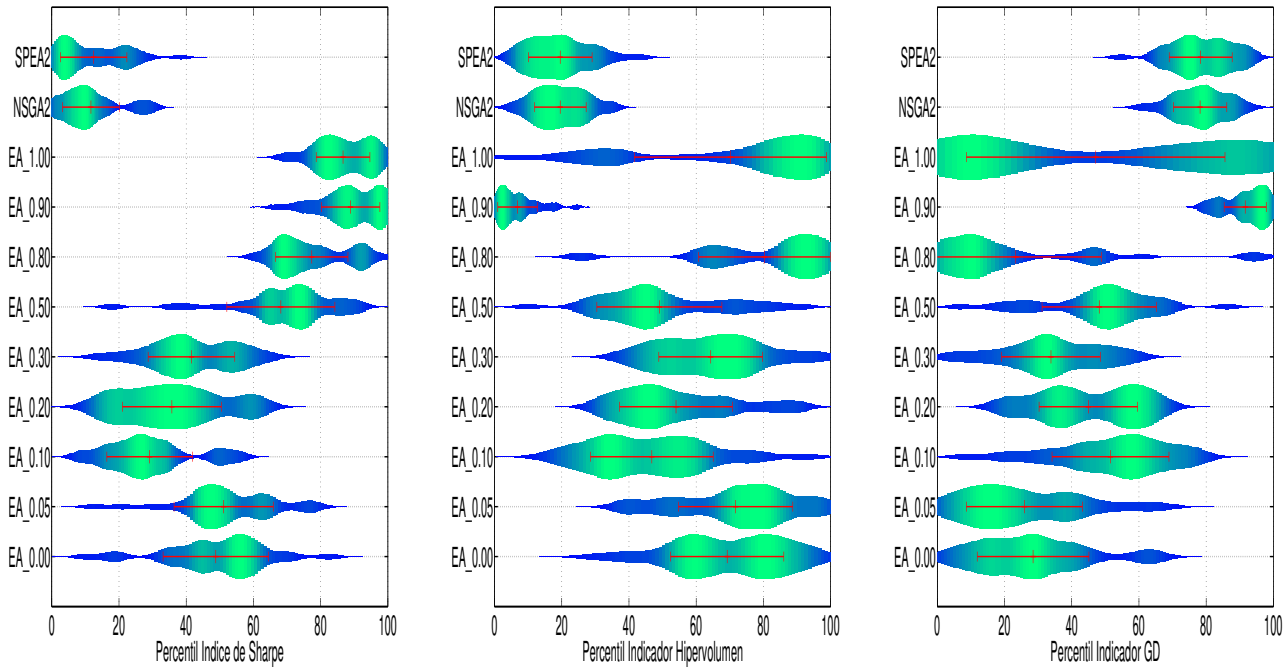


Fig. 1. Distribución de percentiles alcanzada en 30 ejecuciones de los diferentes algoritmos considerados: índice de Sharpe (izquierda), hipervolumen (medio), distancia generacional (derecha) para K=14.

con todos de los algoritmos empleados) como conjunto de referencia. Al ser una medida de distancia al conjunto de referencia, cuanto menor sea el valor de GD, mejor.

B. Resultados Experimentales

La experimentación se estructuró bajo el esquema de realizar simultáneamente un análisis de sensibilidad para la activación del componente de memoria élite con diferentes valores de la probabilidad de activación, y una comparación con tres algoritmos evolutivos conocidos en el área según los supuestos explicados anteriormente en III-A (nótese que para el valor $P_\xi = 0$, el EA con memoria élite es equivalente a IBEA de acuerdo a lo descrito en la Sección II-B). En este apartado experimentamos para encontrar la mejor relación de parametrización para la activación de la memoria élite, para un valor de cardinalidad ($K = 14$) definido como restricción del problema además de la integración con el índice de Sharpe como base de la propuesta evolutiva.

Los resultados se presentan en la Figura 1 desde un punto de vista comparativo. Para ello se ha considerando la agregación de resultados de todos los algoritmos, lo que permite asociar a cada ejecución de cualquiera de los algoritmos su percentil de rendimiento (esto es, cuál sería su escalafón en relación con las demás ejecuciones de la calidad del frente de Pareto generada, o la calidad –según el indice de Sharpe– de la solución seleccionada del mismo). A partir de aquí, se puede calcular el rango de percentiles en el que se mueve cada algoritmo en el conjunto de las ejecuciones del mismo (cuanto más cerca de 100, mejor es el rendimiento comparativo). Intuitivamente, se puede observar que la activación de la memoria de élite proporciona una compensación entre diversificación e intensificación: mientras que los valores altos de P_ξ ejercen una alta presión sobre una región muy localizada del espacio de búsqueda, los valores bajos de P_ξ dan como resultado una presión extendida más suave al realizarse sobre un conjunto más diverso de soluciones. En cualquier caso, es importante recordar que esta presión se ejerce en el ámbito del índice de Sharpe, por lo que el efecto apreciado en las medida de rendimiento multiobjetivo son consecuencia indirecta del sesgo que la memoria élite introduce en el proceso de búsqueda.

Se puede observar cómo valores altos de P_ξ resultan en un mejor desempeño con respecto al indicador de rendimiento financiero (índice de Sharpe) con alta grado de significación estadística con respecto a valores más bajos: el mejor valor mediano se alcanza para $P_\xi = 0.90$, si bien no hay diferencia estadística (p-value $> .1$) con $P_\xi = 1.00$ de acuerdo con un test Wilcoxon ranksum (si la hay a nivel $\alpha = 0.05$ con $P_\xi \leqslant 0.80$). Sin embargo, se debe tener en cuenta un efecto parcialmente opuesto sobre el hipervolumen y la distancia generacional: los mejores valores medianos se alcanzan en este caso para $P_\xi = 0.80$ (la diferencia es significativa con otros valores en el caso de hipervolumen, y solo con $P_\xi \leqslant 0.30$ en distancia generacional, si bien la dispersión de valores es mayor para los valores superiores de P_ξ). Atribuimos

estos resultados a que el algoritmo focaliza en demasía sobre soluciones cuasi óptimas con respecto al índice de Sharpe, y por lo tanto le dificulta extenderse por regiones más grandes del frente de Pareto. También debe tenerse en cuenta no obstante que en todos los casos el EA con memoria élite supera a las contrapartes de ella (NSGA-II, SPEA2 e IBEA).

IV. Conclusiones

La optimización de carteras es un escenario natural para el uso de algoritmos evolutivos multiobjetivo, tanto por la formulación intrínseca del problema como por la capacidad y flexibilidad de este tipo de algoritmos. Una característica particularmente interesante de este problema es la existencia de un indicador natural de eficiencia de la cartera (el índice de Sharpe). Esto tiene dos ventajas principales: por un lado, proporciona un mecanismo apropiado para identificar una solución después del proceso de optimización multiobjetivo; por otro lado, constituye una medida legítima para indicar direcciones de búsqueda preferidas en el espacio multiobjetivo. En este sentido, la ausencia de tales indicaciones es típicamente una de las dificultades que enfrentan los algoritmos meméticos en este tipo de dominios, lo que hace que la búsqueda se enfoque en direcciones elegidas al azar o que se desvíe en diferentes direcciones no necesariamente alineadas con carteras financieramente eficientes. Sin embargo, explotar este indicador puede proporcionar un objetivo razonable para la búsqueda, adaptando así el algoritmo a las especificidades del problema.

Nuestra experimentación con datos de 20 valores tomados de la Bolsa de Valores de Colombia en el período comprendido entre 2010 y 2016 ha brindado apoyo a esta afirmación. El modelo algorítmico considerado incluye el uso de un mecanismo de memoria élite basado en este indicador para modular el perfil de intensificación y diversificación de la búsqueda. Como se muestra en la sección anterior, los resultados del EA son notablemente mejores (desde el punto de vista del rendimiento financiero) que otras propuestas algorítmicas sobre el benchmark descrito, lo que indica que resulta adecuado para enfocar la búsqueda hacia carteras eficientes.

Hay muchas vías para seguir investigando. Sin lugar a dudas, confirmar estos hallazgos en otros conjuntos de datos es uno de ellos. Asimismo, tal como se apuntó en la sección II-B, un enfoque memético clásico vendría determinado por la inclusión de un componente de búsqueda local. Nuestra experimentación con el mismo indica que dicho componente es particularmente útil [19], lo que de manera natural sugiere que el siguiente paso debe ser comprobar la sinergia entre el mecanismo de memoria élite y el componente de búsqueda local. Es precisamente en esta linea en la que estamos dirigiendo los desarrollos actuales.

Agradecimientos

Este trabajo está financiado en parte por el Ministerio Español de Economía y Competitividad (TIN2017-85727-C4-1-P – https://deepbio.wordpress.com/) y la Universidad de Málaga, Campus de Excelencia Internacional Andalucía Tech.

References

[1] J. Philippe, *Valor en riesgo: el nuevo paradigma para el control de riesgos con derivados*. Limusa, 1999.

[2] H. M. Markowitz, "Portfolio selection," *Journal of Finance*, vol. 7, pp. 77–91, 1952.

[3] W. F. Sharpe, "Mutual fund performance," *Journal of Business*, vol. 39, pp. 119–138, 1966.

[4] K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan., "A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II," 2000, kanGAL report 200001, Indian Institute of Technology, Kanpur, India, 2000.

[5] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength pareto evolutionary algorithm," ETH Zürich, Gloriastrasse 35, CH-8092 Zurich, Switzerland, Tech. Rep. 103, 2001.

[6] E. Zitzler and S. Künzli, "Indicator-based selection in multiobjective search," in *Parallel Problem Solving from Nature – PPSN VIII*, ser. Lecture Notes in Comuter Science, X. Yao *et al.*, Eds., vol. 3242. Berlin Heidelberg: Springer-Verlag, 2004, pp. 832–842.

[7] A. M. Zubeldia, L. M. M. Zabalza, and M. Z. Zubiaurre, "El modelo de markowitz en la gestión de carteras," *Cuadernos de Gestión*, vol. 2, no. 1, pp. 33–46, 2002.

[8] D. Wolpert and W. Macready, "No free lunch theorems for optimization," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 67, pp. 67–82, 1997.

[9] P. Moscato and C. Cotta, "An accelerated introduction to memetic algorithms," in *Handbook of Metaheuristics*, ser. International Series in Operations Research & Management Science, M. Gendreau and J.-Y. Potvin, Eds. Springer International Publishing AG, 2019, vol. 272, pp. 275–309.

[10] F. Neri and C. Cotta, "Memetic algorithms and memetic computing optimization: A literature review," *Swarm and Evolutionary Computation*, vol. 2, pp. 1–14, 2012.

[11] F. Neri, C. Cotta, and P. Moscato, Eds., *Handbook of Memetic Algorithms*, ser. Studies in Computational Intelligence. Berlin Heidelberg: Springer-Verlag, 2012, vol. 379.

[12] C. Cotta, L. Mathieson, and P. Moscato, "Memetic algorithms," in *Handbook of Heuristics*, M. Resende, R. Martí, and P. Pardalos, Eds. Cham: Springer International Publishing, 2018, pp. 607–638.

[13] F. E. Colomine Durán, C. Cotta, and A. J. Fernández-Leiva, "A comparative study of multi-objective evolutionary algorithms to optimize the selection of investment portfolios with cardinality constraints," in *Applications of Evolutionary Computation: EvoApplications 2012*, ser. Lecture Notes in Computer Science, C. Di Chio *et al.*, Eds., vol. 7248. Berlin, Heidelberg: Springer, 2012, pp. 165–173.

[14] C. Houck, J. Joines, M. Kay, and J. Wilson, "Empirical investigation of the benefits of partial lamarckianism," *Evolutionary Computation*, vol. 5, no. 1, pp. 31–60, 1997.

[15] B. Colombian Stock Market, "Mercado de renta variable [accessed november 2017]," https://www.bvc.com.co/pps/tibco/portalbvc, 2008.

[16] S. Bleuler, M. Laumanns, L. Thiele, and E. Zitzler, "PISA—A Platform and Programming Language Independent Interface for Search Algorithms," in *Conference on Evolutionary Multi-Criterion Optimization (EMO 2003)*, ser. Lecture Notes in Computer Science, C. M. Fonseca, P. J. Fleming, E. Zitzler, K. Deb, and L. Thiele, Eds., vol. 2632. Berlin: Springer, 2003, pp. 494–508.

[17] E. Zitzler and L. Thiele, "Multiobjective optimization using evolutionary algorithms — A comparative case study," in *Parallel Problem Solving from Nature — PPSN V*, ser. Lecture Notes in Computer Science, A. E. Eiben, T. Bäck, M. Schoenauer, and H.-P. Schwefel, Eds., vol. 1498. Berlin, Heidelberg: Springer Berlin Heidelberg, 1998, pp. 292–301.

[18] D. A. V. Veldhuizen and G. B. Lamont, "Multiobjective evolutionary algorithms: Analyzing the state-of-the-art," *Evol. Comput.*, vol. 8, no. 2, pp. 125–147, 2000.

[19] F. Colomine Durán, C. Cotta, and A. Fernández-Leiva, "Sensitivity to partial lamarckism in a memetic algorithm for constrained portfolio optimization," in *Evostar 2021 Late-Breaking Abstracts*, A. Mora, Ed., 2021.

Un Enfoque Evolutivo Multi-Operador para el Diseño de Rutas Turísticas con Nivel de Interés Dependiente del Tiempo.

David A. Pelta
Depto. de Ciencias de la Comp. e IA
Universidad de Granada
Email: dpelta@ugr.es

Juan Carlos Ortegón
Grupo de Trabajo en
Modelos de Decisión y Optimización
Universidad de Granada
Email: homer@thesimpsons.com

Marcelino Cabrera
Depto. de Leng. y Sist. Informáticos
Universidad de Granada
Email: mcabrera@ugr.es

***Resumen*—Partiendo del hecho de que el nivel de interés de un monumento, parque, mirador, etc. depende del momento en que se visite, en este trabajo se propone una versión tiempo-dependiente del problema de diseño de rutas turísticas con puntos de interés.**

Se aborda su resolución mediante un algoritmo evolutivo (AE) basado solamente en mutaciones y se exploran diferentes operadores, incluyendo un operador de mutación Macro (basado en la selección probabilista de 4 operadores simples).

El AE con el operador Macro permite obtener rutas con valores más altos de interés que los obtenidos cuando el mismo AE utiliza cada uno de los operadores simples. La adaptación dinámica de las probabilidades de selección es la clave para estos resultados.

En este tipo de problemas, donde intervienen usuarios con sus preferencias y donde abunda la imprecisión y/o incertidumbre (respecto a tiempos de viaje, tiempos de visita, etc.) es fundamental resaltar el rol de las metaheurísticas como herramientas generadoras de alternativas que deben analizarse más allá de los valores de la función objetivo.

I. Introducción

El problema de diseño de rutas turísticas con puntos de interés (tourist trip design problem) (TTDP)) personalizadas involucra dos aspectos esenciales: la selección de puntos de interés adecuados a las preferencias del turista, y la generación de los itinerarios para recorrer dichos puntos (todos o un subconjunto de ellos). Mientras que el primer aspecto suele relacionarse con sistemas de recomendación, que deben identificar las preferencias del usuario, el segundo está relacionado con problemas de decisión y optimización.

En general, para el diseño de los itinerarios, los modelos de TTDP toman como base los bien conocidos *Orienteering Problem* (OP), *Team Orienteering Problem* (TOP) y sus variantes ([5], [6], [8], [10], [1], [3], [9].

En los últimos años han surgido variantes interesantes del problema. Por ejemplo, en [2] abordan una versión del TTDP donde los puntos de interés se agrupan en clusters que representan diferentes tipos de sitios de atracción. Además, se imponen límites mínimos/máximos al número de puntos de interés pertenecientes a cada clúster que se visitan en la misma ruta. Para su resolución se propone un GRASP difuso.

En [4], se estudia una variante del TOP, cuya principal característica es que el "score" obtenido en un vértice visitado depende de la duración de la visita en ese vértice. El problema se resuelve mediante una búsqueda local iterativa.

Finalmente, en [7] proponen una versión extendida del TTDP basada en dos conceptos adicionales. El primero es la consideración de múltiples turistas, donde se tienen en cuenta sus preferencias individuales sobre los puntos de interés, y el segundo es la introducción del concepto de relación social mutua entre los diferentes turistas.

En este trabajo se plantean dos objetivos. En primer lugar, se propone una extensión del TTDP basada en el siguiente hecho: *la franja horaria en la cual se visita un punto condiciona su nivel de interés*. Por ejemplo visitar los Jardines del Triunfo en Granada es mucho más interesante por la noche, cuando las fuentes de agua están encendidas e iluminadas, que hacerlo por la mañana.

El problema de diseño de rutas turísticas con puntos cuyo interés depende del tiempo (TTDP-TDS, *tourist trip design problem with time dependent scores*) parte de un conjunto finito de puntos de interés (POIs). Cada POI tiene asociado un tiempo de visita, un nivel de interés y un factor de recomendación asociado a cada franja horaria. Se tiene además el tiempo de desplazamiento entre cada par de POIs. El objetivo en el TTDP-TDS es maximizar el interés total obtenido sujeto a un tiempo máximo de visita disponible.

En [11] se ha abordado una variante similar: un TOP con ventanas de tiempo, añadiendo como elemento relevante que la puntuación de la visita a una atracción es diferente en función de la hora de la visita. Sin embargo, los resultados mostrados generan dudas sobre su factibilidad, las instancias del problema no están disponible y el intento de contacto con los autores ha resultado infructuoso.

En segundo lugar, se propone un algoritmo evolutivo (AE) para su resolución. Vale la pena mencionar que el AE no utiliza cruce y se proponen 5 operadores diferentes de mutación, estando uno de ellos basado en la selección probabilística de los otros cuatro. Aprovechando que se trata de un algoritmo basado en población, se analiza el tipo de información que se

puede obtener del conjunto de soluciones obtenidas, más allá de los valores de la función objetivo. Todos los experimentos se realizan sobre datos de la ciudad de Granada.

A partir de aquí, el artículo está organizado de la siguiente manera. En la sección II se describen las características principales de la nueva versión del TTDP así como el procedimiento realizado para generar las instancias de prueba. Luego, en la sección III se presenta el algoritmo evolutivo propuesto, decribiendo los diferentes operadores de mutación. La sección IV se dedica a los experimentos y resultados. Finalmente, las conclusiones del trabajo se describen en la sección V.

II. Descripción del Problema

El problema de diseño de rutas turísticas con puntos cuyo interés depende del tiempo (TTDP-TDS, *tourist trip design problem with time dependent scores*) parte de un conjunto N de nodos o puntos de interés (POIs). Cada nodo $i \in N$ tiene asociado un interés o score S_i y varios factores de recomendación f_{it} para ponderar el interés de acuerdo con el momento en el que se visite el nodo. El tiempo de desplazamiento desde el nodo i al nodo j viene dado por t_{ij}. Por otra parte, no todos los nodos del conjunto pueden ser visitados debido a que existe un máximo tiempo disponible (T_{max}). Existe un nodo $n_{ini} \in N$ que sirve como punto inicial y final de la ruta. El problema se modeliza de la siguiente forma.

Parámetros

- N - Conjunto de nodos.
- n_{ini} - indice del nodo inicial/final del recorrido
- T - Número de franjas horarias.
- S_i - interés o score de cada nodo.
- t_{ij} - Tiempo de viaje entre el nodo i y el nodo j.
- f_{it} - Factor de recomendación de visitar el nodo i en el periodo t.
- v_i - Tiempo requerido para visitar el nodo i (tiempo de servicio).
- T_{max} - Máximo tiempo disponible para el recorrido.

Representación de la solución

Se definen y_{it}, como una variable binaria que vale 1 si el nodo i es visitado en el periodo de tiempo t, 0 en otro caso; y π como una permutación de k elementos (la long. de la ruta) donde $\pi(i)$ indica el nodo que se visita en la posición i de la ruta. Esta permutación permite calcular el tiempo de la ruta.

Función Objetivo

$$Maximizar \quad \sum_{i \in N \setminus \{n_{ini}\}} \sum_{t \in T} S_i \cdot f_{it} \cdot y_{it} \qquad (1)$$

sujeto a:

$$T_{visita} + T_{viaje} \leq T_{max} \qquad (2)$$

donde

$$T_{visita} = \sum_{i \in N \setminus \{n_{ini}\}} \sum_{t \in T} v_i \cdot y_{it} \qquad (3)$$

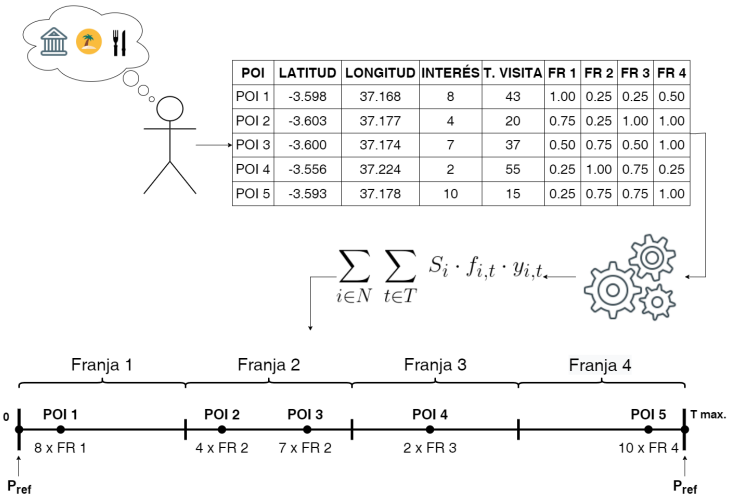


Figura 1. Rutas de puntos con interés tiempo-dependientes

$$T_{viaje} = t_{ini,\pi(1)} + \left(\sum_{i=1}^{k-1} t_{\pi(i),\pi(i+1)} \right) + t_{\pi(k),ini} \qquad (4)$$

La función objetivo (1) maximiza la suma total de la puntuación obtenida. Aunque la visita a un punto puede abarcar dos franjas horarias diferentes, en esta versión del problema se considera que el interés que aporta un nodo a la puntuación está dado por el momento en el que se comienza su visita. Es decir, si un nodo i se comienza a visitar en la franja horaria t, la contribución que aporta al interés total es $S_i \times f_{it}$.

La restricción (2) nos asegura que el tiempo total de la ruta generada (tiempos de visita más tiempos de desplazamientos) no supera el tiempo máximo disponible T_{max}.

La Fig. 1 muestra un ejemplo de solución y el cálculo de la función objetivo.

La solución consta de 5 puntos de interés. Partiendo del P_{ref} (por ejemplo, un hotel), el turista se desplaza al POI_1. La llegada al punto se produce en la primera franja horaria así que el interés del POI es $S_1 \times f_{11} = 8 \times 1 = 8$. Luego se desplaza al POI_2, llegando en la segunda franja horaria y aportando un interés de $4 \times 0{,}25$. El proceso continúa hasta completar el tiempo de visita del POI_5 y regresar al punto de inicio P_{ref}.

II-A. Generación de instancias del problema

De cara a facilitar la reproducibilidad de los resultados, se describe a continuación el procedimiento de generación de instancias, las cuales se pueden encontrar en http://18.156.111.23/TTDP-TDS/web/.

Los datos del mapa, puntos de interés y tiempos de desplazamiento a pie se han obtenido mediante el paquete de Python **OSMnx**[1] que permite descargar datos espaciales de OpenStreetMap y modelar, proyectar, visualizar y analizar redes de calles del mundo real.

[1](https://osmnx.readthedocs.io/en/stable/)

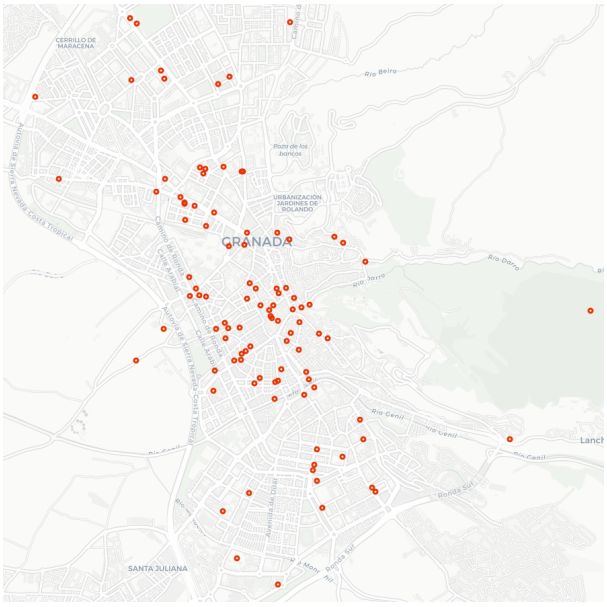


Figura 2. Mapa de Granada y conjunto base de puntos de interés

Para la obtención de los POIs, se utiliza la función de OSMNx *geometries_from_place('Granada, Spain')* que devuelve 1000 puntos de interés (incluyendo museos, bares, restaurantes, sitios religiosos, etc.).

A partir de esos 1000 puntos se seleccionan aleatoriamente 100 puntos para los cuales se completan con valores aleatorios los datos faltantes. La distribución de los mismos en Granada se muestran en la Fig. 2.

Se consideran 4 franjas horarias ($T = 4$) y un valor $T_{max} = 480$. Cada franja horaria tiene una duración de 120 minutos. Estos valores se mantienen fijos para todas las instancias. Finalmente, para cada POI se tiene:

- **osmid:** identificador único del POI.
- **name:** nombre del POI.
- **amenity:** servicio que ofrece.
- **lat:** latitud.
- **lng:** longitud.
- **TiempoVisita** v_i**:** tiempo de visita (minutos). Valor entero generado al azar en el intervalo $[20, 60]$.
- **Interés** S_i**:** interés del punto. Valor entero generado al azar en el intervalo $[1, 10]$.
- **FactoresRecom.** $f_{ij}, j = 1, 2, 3, 4$ Factores de recomendación. Los valores $\{0{,}25; 0{,}50; 0{,}75; 1{,}00\}$ se asignan aleatoriamente entre los f_{ij}. En todos los casos existe i tal que $f_{ij} = 1$ para garantizar que existe una franja "ideal" para visitar el POI.
- **near_node:** Nodo del grafo asociado al POI.

Finalmente, se generan 10 instancias de 51 nodos (elegidos aleatoriamente sobre los 100 anteriores). El primer nodo de cada instancia es n_{ini}: el punto de origen y llegada de la ruta.

III. Propuesta de Algoritmo Evolutivo

Para la resolución del problema se propone un algoritmo evolutivo sin operador de cruce. El esquema se muestra en el Algoritmo 1.

Cada individuo de la población inicial se genera utilizando un procedimiento greedy aleatorizado. La función `generarPoblacionInicial`$(popSize)$ construye cada individuo de la siguiente manera. Se selecciona aleatoriamente un POI inicial. Posteriormente se construye una lista de POIs candidatos y se elige al azar un elemento de la lista. el cual se incorpora a la solución. El proceso se repite mientras no se sobrepase el tiempo máximo de ruta.

En cada generación, de la población actual de $popSize$ individuos se eligen K padres mediante selección por ruleta que se incorporan a una población intermedia. Mediante mutación, de cada padre se genera una solución hija. Si la nueva solución es factible y mejor que el padre, se incorpora a la población intermedia.

Finalmente, se ordena la población intermedia y se cogen los mejores $popSize$ elementos para reemplazar a la población original.

Algorithm 1: Algoritmo Evolutivo propuesto.

```
/* popActual, auxPop población actual e intermedia;
popSize, K tamaños de las poblaciones;
maxGens límite de generaciones */;
popActual = generarPoblacionInicial(popSize);
numGen = 0;
while numGen < MaxGens do
    generar poblacion intermedia auxPop;
    k = 0;
    auxPop = {};
    repeat
        padre = seleccionPorRuleta(popActual);
        auxPop = auxPop ∪ {padre};
        hijo = mutacion(padre);
        if hijo es factible y hijo mejor que padre)
         then
            auxPop = auxPop ∪ {hijo} ;
        end
        k++
    until k < K;
    // Actualización de Pop
    popActual = ordenarYtruncar(auxPop);
    numGen++;
end
```

El único elemento disponible para generar nuevas soluciones es el operador de mutación. En este trabajo, teniendo en cuenta que una solución se representa mediante un vector de enteros de longitud variable (un subconjunto de los puntos de interés disponible), se proponen los siguientes operadores de mutación:

1. **Desplazamiento**: se eligen dos posiciones i, j. Luego, se desplaza el punto de la posición j a la posición $i-1$
2. **Intercambio**: se eligen dos posiciones i, j. Luego, se intercambian los puntos correspondientes.
3. **Inversión**: se eligen dos posiciones i, j. Luego se invierte el recorrido entre las dos posiciones.
4. **Inserción**: se elige una posición i. Luego, se elige un punto entre los POIs aún no visitados y se inserta en la posición i.
5. **Macro**: Se selecciona uno de los cuatro operadores anteriores con una probabilidad proporcional a su "éxito". Si un operador da lugar a una solución mejor que la original, se le asigna un punto de crédito.

El algoritmo propuesto siempre opera con soluciones factibles. Por tanto, las soluciones no factibles obtenidas por mutación, se descartan.

IV. Experimentos y Resultados

Definimos como AE_i a la variante del algoritmo evolutivo que utiliza el operador $i = 1, 2, \ldots, 5$. Para cada AE_i y para cada una de las 10 instancias del problema, se realizan 30 ejecuciones.

Los experimentos están orientados a:

1) analizar el comportamiento de las diferentes variantes del algoritmo. 2) mostrar el tipo de información que se puede extraer de las soluciones, mas allá de su valor de *fitness*.

Todas las variantes del algoritmo utilizan los siguientes parámetros:

- **Tamaño población**: 30
- **Máximo tiempo de visita**: 480 minutos
- **Número de generaciones**: 50
- **Número de hijos por generación**: 50

IV-A. Análisis de los AE_i

El objetivo de este experimento es evaluar y comparar las variantes del algoritmo evolutivo utilizando cada uno de los operadores de mutación.

La Tabla I muestra la media y desviación típica obtenidas con cada AE_i en cada instancia de prueba. La última fila muestra los totales calculados sobre todas las instancias.

Se observa claramente que el AE_4 (operador de inserción) y el AE_5 (Macro) obtienen los mejores resultados. En todas las instancias, el AE_5 alcanza los valores medios más altos. Los resultados de AE_1, AE_2 y AE_3 son muy similares.

Los test de Mann-Whitney y Kolmogorov-Smirnov indican que las diferencias en nivel de interés entre AE_4 y AE_5 son estadísticamente significativas (a favor de este último).

Sin embargo, dado que disponemos de un conjunto de soluciones, sería un desperdicio descartar otra información que se puede extraer de las mismas. Otro aspecto relevante a estudiar es el tiempo promedio de visita de las soluciones obtenidas. La Tabla II muestra las medias y desviaciones típicas (sobre 30 ejecuciones) del tiempo de visita de las rutas sobre cada problema y variante del algoritmo.

Recordando en primer lugar que $T_{max} = 480$, se observa que las variantes AE_1, AE_2 y AE_3 utilizan prácticamente todo el tiempo disponible (aprox. 470 minutos en promedio) mientras AE_4 y AE_5 utilizan alrededor de 400 y 435 minutos aproximadamente.

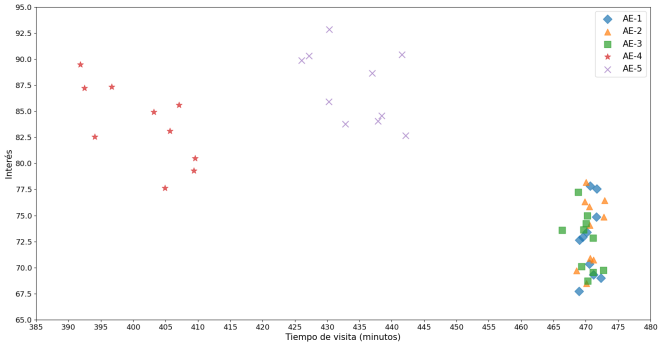


Figura 3. Relación entre nivel de interés promedio y tiempo de las rutas obtenidas por cada variante.

Sin embargo, el nivel de interés de las rutas proporcionadas por las tres primeras variantes es significativamente peor que el obtenido por las dos últimas.

Esta relación entre ambas medidas (interés vs. tiempo) puede verse claramente en la Fig. 3. Cada punto representa el promedio de interés/tiempo obtenido (sobre 30 ejecuciones) por cada variante del algoritmo en cada instancia. Los puntos de abajo a la derecha corresponden a las variantes AE_1, AE_2 y AE_3. Por definición, los operadores de mutación asociados a estas variantes solamente reorganizan los POIs existentes en la ruta, sin agregar ni eliminar nuevos puntos. Las variantes AE_4 y AE_5, permiten incorporar nuevos puntos a la solución, y por tanto mejorar el nivel de interés de las mismas.

Las soluciones obtenidas por AE_5 reflejan un mejor equilibrio entre interés y tiempo de visita: alcanzan el mejor interés, dejando un margen de tiempo relevante para gestionar las posibles demoras/adelantos producidas en los tiempos de visita o desplazamientos entre puntos.

Esta ventaja también se observa en los tiempos de ejecución de cada AE_i, mostrados en la Tabla III. El algoritmo AE_4 (utilizando Inserción) permite obtener buenos resultados pero consume casi 3.5 veces más tiempo que las variantes mas sencillas y el doble que AE_5. En cambio, esta última consume el doble de tiempo que las variantes sencillas y la mitad que AE_4.

IV-B. Frecuencia de uso de los operadores

El operador de mutación *Macro* realiza una selección probabilista sobre los operadores básicos disponibles. Para ver la evolución de dichas probabilidades se ejecutó dos veces el algoritmo AE_5 durante 100 generaciones, registrando las probabilidades de selección de cada operador en cada generación. Ambas ejecuciones se muestran en la Fig. 4.

Aunque no es posible establecer conclusiones generales a partir de estos ejemplos ilustrativos, sí se puede observar claramente como se produce una adaptación dinámica de las probabilidades en función del éxito de los operadores seleccionados. Con el transcurso de las generaciones, los operadores

	Algoritmo									
	AE_1		AE_2		AE_3		AE_4		AE_5	
Prob.	Prom.	dt	Prom.	dt	Prom.	dt	Prom.	dt	Prom.	dt
1	72,65	5,69	74,88	5,33	72,83	4,49	82,55	3,56	85,93	5,84
2	77,56	6,34	76,45	5,11	77,25	5,19	84,92	3,77	90,43	4,92
3	77,82	4,85	78,19	7,58	74,98	6,59	89,50	4,38	92,84	6,06
4	72,92	6,44	75,85	5,97	73,65	4,76	87,33	3,11	89,88	4,86
5	68,98	5,35	69,71	6,14	68,73	4,91	77,63	4,43	82,68	5,99
6	74,87	5,22	76,33	4,61	74,22	4,97	85,62	3,44	88,64	4,41
7	67,73	5,17	70,73	5,70	69,57	5,83	80,48	3,48	83,77	4,29
8	70,34	5,91	70,91	4,69	70,12	5,13	83,10	3,15	84,06	5,45
9	69,32	4,49	68,48	5,26	69,76	6,27	79,32	3,81	84,56	3,82
10	73,40	4,92	74,09	6,74	73,61	5,87	87,23	3,51	90,31	6,05
Total	72,56	6,34	73,56	6,52	72,47	5,98	83,77	5,15	87,31	6,14

Cuadro I
MEDIA Y DESVIACIÓN TÍPICA (SOBRE 30 EJECUCIONES) DEL INTERÉS SOBRE CADA PROBLEMA Y OPERADOR.

	Algoritmo									
	AE_1		AE_2		AE_3		AE_4		AE_5	
Prob.	Prom.	dt	Prom.	dt	Prom.	dt	Prom.	dt	Prom.	dt
1	469,02	5,35	472,81	6,03	471,10	6,97	394,07	28,74	430,29	25,66
2	471,70	7,36	472,91	5,52	468,85	7,74	403,19	19,26	441,58	20,40
3	470,68	4,86	470,04	6,68	470,21	7,13	391,84	24,97	430,34	27,05
4	469,56	8,26	470,54	8,44	469,66	8,96	396,67	21,74	426,04	21,34
5	472,34	6,36	468,60	7,87	470,29	5,91	404,91	25,96	442,11	16,80
6	471,60	6,32	469,84	7,80	470,02	8,12	407,10	17,92	436,98	20,64
7	468,97	7,24	471,20	6,00	471,11	7,38	409,60	28,23	432,81	23,06
8	470,53	6,85	470,68	6,89	469,31	6,18	405,68	20,91	437,87	24,98
9	471,22	6,69	470,13	6,78	472,69	6,52	409,37	23,78	438,46	15,52
10	470,18	7,17	470,62	7,18	466,31	11,28	392,45	27,90	427,19	21,91
Total	470,58	6,70	470,74	6,98	469,96	7,81	401,49	24,77	434,37	22,37

Cuadro II
MEDIA Y DESVIACIÓN TÍPICA (SOBRE 30 EJECUCIONES) DEL TIEMPO DE LA RUTA SOBRE CADA PROBLEMA Y OPERADOR.

	Algoritmo				
Prob	AE_1	AE_2	AE_3	AE_4	AE_5
1	8,75	8,77	8,74	30,17	15,54
2	8,70	8,80	8,74	30,36	15,42
3	8,88	8,77	8,82	30,44	15,14
4	8,95	9,06	8,94	30,07	15,61
5	8,57	8,41	8,44	29,64	13,50
6	8,36	8,62	8,50	29,63	14,82
7	8,73	8,84	8,76	30,05	16,74
8	8,73	8,93	8,85	30,85	14,92
9	9,00	8,84	9,04	30,88	15,24
10	8,69	8,67	8,73	30,24	15,90
Total	8,73	8,77	8,76	30,23	15,29

Cuadro III
TIEMPO MEDIO DE UNA EJECUCIÓN DE CADA ALGORITMO EN CADA INSTANCIA (EN SEGUNDOS).

2 (Intercambio) y 4 (Inserción) aumentan su probabilidad de selección. El operador 3 (inversión), al no conseguir mejoras, disminuye rápidamente su probabilidad.

El punto a partir del cual las probabilidades no varían, indica que el algoritmo no es capaz de mejorar ninguna de las soluciones de la población.

IV-C. Sobre las soluciones

El último análisis está relacionado con otra característica de las soluciones obtenidas: la frecuencia de aparición de los POIs.

Los resultados aparecen en la Fig. 5. A partir de las 30 mejores soluciones asociadas a la instancia 1 (independientemente del AE_i utilizado), se ha calculado la frecuencia de aparición de cada uno de los 100 POIs del conjunto N en dichas soluciones. La tabla muestra solamente aquellos puntos que han aparecido, al menos, en 6 soluciones.

Destacan los nodos 30, 8, 13, 4, 10, 26 y 15 que aparecen en más de 20 sobre 30 soluciones. En otras palabras, son POIs que deben ser considerados en el diseño de las rutas.

Todas las soluciones analizadas tienen 13 o 14 POIs. La Fig. 5 también muestra la frecuencia de aparición por posición en la ruta de cada POI. A mayor intensidad en el color, mayor frecuencia. En general, se puede observar que la mayor parte de las ocurrencias de cada POI se concentra en unas pocas posiciones. Por ejemplo, el primer nodo (nodo 30), aparece casi siempre entre las posiciones 4 y 7 de la ruta. El nodo 8, casi siempre aparece en las tres primeras posiciones, mientras que el nodo 4 se ha visitado casi siempre en las posiciones finales de la ruta. Naturalmente esta distribución está asociada

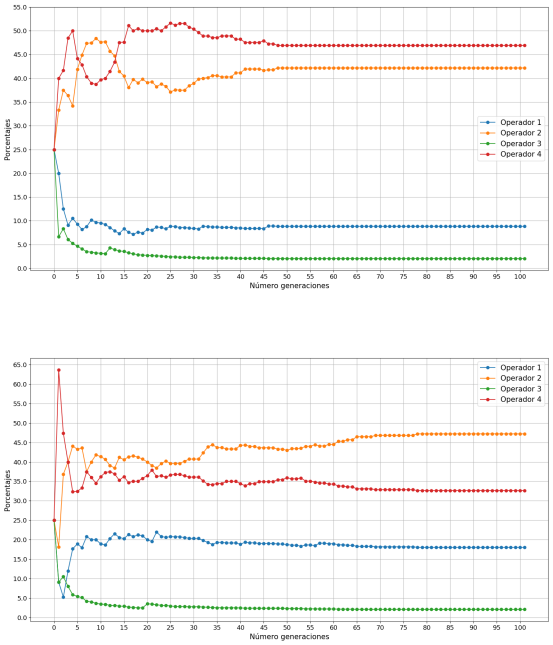


Figura 4. Evolución de las frecuencias de selección de cada operador en MacroMutacion. Se muestran dos ejecuciones diferentes de 100 generaciones.

		Frecuencia por posición													
Nodo	Frecuencia	1	2	3	4	5	6	7	8	9	10	11	12	13	14
30	29	2	1	2	8	4	5	6	1						
8	27	5	11	6	1		1	1		1		1			
13	26				2	9	10	5							
4	24			1				1	3	6	6	4	3		
10	24			1		1		2	1	6	5	7	1		1
26	23				5	8	4	3	2				1		
15	21	1	3	6	2	1	1	2	3	2					
47	19								4	5	7	1	2		
12	16	1	1						8	2	2	2			
6	14	1	1				2		2	1	1	1	3	2	
16	13					1		1		1	3	3	3	1	
25	13	4	4	2	2									1	
42	11	1	1	2	4	3									
48	10		1	2	2							2	1	2	
9	9						1	1				2	4	1	
27	9									2	5	1	1		
43	9	1				1	1	3	2			1			
36	8		1	2		1	2	1		1					1
41	8	4	2	2											
37	6	3	1	1	1										

Figura 5. Frecuencia de aparición total y por posición de cada POI en las mejores 30 soluciones de la instancia 1.

a que el nivel de interés del punto depende de la franja horaria en la cual se visita.

V. Conclusiones

En este trabajo se ha presentado una versión del TTDP donde el nivel de interés de un POI depende del momento en que se realice la visita.

Para su resolución se ha propuesto un algoritmo evolutivo basado solamente en mutaciones. Los resultados obtenidos muestran que el AE que utiliza un operador de mutación Macro (basado en la selección probabilista de 4 operadores individuales) permitió obtener rutas con valores más altos de interés que los obtenidos cuando el mismo AE utiliza cada uno de los operadores simples. La adaptación dinámica de las probabilidades de selección es la clave para estos resultados.

Sobre el conjunto de soluciones obtenidas por las diferentes versiones del AE, se observó también que el AE con el operador Macro obtiene soluciones con un mejor balance entre nivel de interés y margen de flexibilidad (en términos del tiempo disponible) lo que permite a su vez, considerar tales soluciones como más realistas de cara a ser realizadas sobre el terreno.

Finalmente, el análisis de frecuencia de uso de cada POI permite detectar cuales son aquellos "indispensables" y además, dar cierta idea sobre cual es el momento más idóneo para visitarlos.

En este tipo de problemas, donde intervienen usuarios con sus preferencias y donde abunda la imprecisión (respecto a tiempos de viaje, tiempos de visita, etc.) es fundamental resaltar el rol de las metaheurísticas como herramientas generadoras de alternativas que deben analizarse más allá de los valores de la función objetivo. Es esta la razón fundamental de la no utilización de operadores más complejos y algoritmos más sofisticados.

Agradecimientos

Los autores agradecen el apoyo del proyecto (TIN2017-86647-P) del Ministerio de de Economía, Industria y Competitividad, España (incluyendo fondos FEDER).

Referencias

[1] R. Deitch and S. P. Ladany. The one-period bus touring problem: Solved by an effective heuristic for the orienteering tour problem and improvement algorithm. *European Journal of Operational Research*, 127(1):69–77, 2000.

[2] A. Expósito, S. Mancini, J. Brito, and J. A. Moreno. A fuzzy grasp for the tourist trip design with clustered pois. *Expert Systems with Applications*, 127:210–227, 2019.

[3] A. Garcia, P. Vansteenwegen, O. Arbelaitz, W. Souffriau, and M. T. Linaza. Integrating public transportation in personalised electronic tourist guides. *Computers & Operations Research*, 40(3):758–774, 2013. Transport Scheduling.

[4] A. Gunawan, K. M. Ng, G. Kendall, and J. Lai. An iterated local search algorithm for the team orienteering problem with variable profits. *Engineering Optimization*, 50(7):1148–1163, 2018.

[5] M. Schilde, K. F. Doerner, R. F. Hartl, and G. Kiechle. Metaheuristics for the bi-objective orienteering problem. *Swarm Intelligence*, 3(3):179–201, 2009.

[6] W. Souffriau, P. Vansteenwegen, J. Vertommen, G. V. Berghe, and D. V. Oudheusden. A Personalized Tourist Trip Design Algorithm for Mobile Tourist Guides. *Applied Artificial Intelligence*, 22(10):964–985, Oct. 2008.

[7] K. Sylejmani, J. Dorn, and N. Musliu. Planning the trip itinerary for tourist groups. *Information Technology & Tourism*, 17(3):275–314, 2017.

[8] P. Vansteenwegen and D. V. Oudheusden. The mobile tourist guide: An OR opportunity. *OR Insight*, 20(3):21–27, 2007.

[9] P. Vansteenwegen, W. Souffriau, G. V. Berghe, and D. V. Oudheusden. The city trip planner: An expert system for tourists. *Expert Systems with Applications*, 38(6):6540–6546, 2011.

[10] X. Wang, B. L. Golden, and E. A. Wasil. Using a genetic algorithm to solve the generalized orienteering problem. In *Operations Research/Computer Science Interfaces*, pp. 263–274. Springer US.

[11] V. F. Yu, P. Jewpanya, S.-W. Lin, and A. P. Redi. Team orienteering problem with time windows and time-dependent scores. *Computers & Industrial Engineering*, 127:213–224, 2019.

Cooperación Inter-algorítmica Multiobjetivo para Reconstrucción Filogenética

Sergio Santander-Jiménez
Dept. Tecnología de los Computadores y de las Comunicaciones,
Universidad de Extremadura
Cáceres 10003, España
sesaji@unex.es

Miguel A. Vega-Rodríguez
Dept. Tecnología de los Computadores y de las Comunicaciones,
Universidad de Extremadura
Cáceres 10003, España
mavega@unex.es

Leonel Sousa
INESC-ID,
Instituto Superior Técnico,
Universidade de Lisboa
Lisboa 1000-029, Portugal
las@inesc-id.pt

***Resumen*—Las técnicas de cooperación inter-algorítmica representan una de las aproximaciones más significativas para potenciar las capacidades de búsqueda de los algoritmos evolutivos. No obstante, la complejidad creciente de los problemas reales da lugar a la necesidad de definir nuevos diseños cooperativos, con estrategias basadas en la monitorización de rendimiento, para mejorar la calidad de las soluciones. Este trabajo explora la cooperación de algoritmos representativos de las tres tendencias fundamentales en optimización multiobjetivo (rankings Pareto, indicadores y descomposición) para resolver el problema de la inferencia de hipótesis filogenéticas. El método estudiado incorpora un elemento supervisor denominado isla élite que, aparte de gestionar las migraciones, evalúa el rendimiento de cada componente a fin de lanzar instancias adicionales de la tendencia algorítmica más satisfactoria en cada etapa de ejecución. Los resultados obtenidos en cinco instancias reales del problema tratado muestran los beneficios de la propuesta a la hora de abordar tareas complejas de optimización, en comparación a los algoritmos individuales y otros modelos de cooperación.**

***Index Terms*—Computación Evolutiva, Optimización Multiobjetivo, Cooperación Inter-algorítmica, Bioinformática.**

I. Introducción

La resolución de problemas de optimización juega un papel fundamental en una amplia variedad de disciplinas científicas. Esta cuestión se ha convertido hoy en día en un reto significativo debido a la complejidad creciente de los problemas a tratar. Factores tales como la disponibilidad de conjuntos de datos grandes, el crecimiento exponencial de los espacios de decisión, la necesidad de considerar modelos matemáticos cada vez más precisos y la optimización multiobjetivo gobiernan la dificultad de estos problemas en contextos reales. Un ejemplo de dominio de aplicación afectado por estas cuestiones de complejidad es la bioinformática, donde es una constante lidiar con datos difíciles de tratar y con conflictos asociados a la definición de múltiples criterios de optimalidad [1].

Los métodos de cooperación inter-algorítmica representan una respuesta a las demandas de optimización robusta de estos problemas. Estas aproximaciones están orientadas a mejorar la calidad de las soluciones empleando esquemas algorítmicos que coordinan la acción conjunta de múltiples algoritmos evolutivos. Ejemplos como la selección de características en entornos no supervisados [2], la planificación multiobjetivo [3] y la optimización de interfaces cerebro-computador [4] dan cuenta de la bondad de estas aproximaciones. No obstante, los modelos cooperativos estándar pueden presentar dificultades a la hora de tratar problemas muy complejos, mostrando problemas potenciales de convergencia o desequilibrios explotación-exploración [5]. Es por ello que es necesaria la definición de diseños cooperativos mejorados para abordar la dificultad creciente de los problemas de optimización actuales.

Este artículo pretende explorar estrategias de cooperación multiobjetivo en un problema bioinformático importante: la inferencia de relaciones filogenéticas entre especies a partir de datos de aminoácidos [6]. Examinaremos la cooperación de tres algoritmos de referencia en el campo de la optimización multiobjetivo: 1) *Non-dominated Sorting Genetic Algorithm II* (NSGA-II) [7]; 2) *Indicator-Based Evolutionary Algorithm* (IBEA) [8]; y el *Multiobjective Evolutionary Algorithm based on Decomposition* (MOEA/D) [9]. Esta selección supone la definición de un modelo inter-algorítmico fundamentado en las tres corrientes principales de diseño en optimización multiobjetivo (rankings Pareto y densidad, selección basada en indicadores de calidad y descomposición), permitiendo la búsqueda paralela y la obtención de información sobre el proceso de optimización desde dichas tres perspectivas.

La idea clave radica en la integración de un cuarto componente que actúa como director de orquesta, monitorizando el rendimiento obtenido por cada tendencia multiobjetivo y ejecutando instancias adicionales del algoritmo que obtuvo el rendimiento más satisfactorio en cada etapa de ejecución. La propuesta, denominada *Inter-Algorithm Cooperation with Elite Island* (IACEI), se fundamenta por tanto en la captura de feedback de cada tendencia multiobjetivo para definir un modelo adaptativo de cooperación y competición destinado a mejorar la robustez del proceso de búsqueda de soluciones.

Este artículo está organizado del siguiente modo. La siguiente sección introduce las características del problema bioinformático a tratar. La Sección III detalla el modelo inter-

Este trabajo ha sido parcialmente financiado por el MCIU (Ministerio de Ciencia, Innovación y Universidades, España), la AEI (Agencia Estatal de Investigación, España) y el FEDER (Fondo Europeo de Desarrollo Regional, Unión Europea), bajo el proyecto PID2019-107299GB-I00/AEI/10.13039/501100011033 (proyecto Multi-HPC-Bio). Este trabajo también ha sido parcialmente financiado por la Junta de Extremadura y la Unión Europea (FEDER) gracias a la ayuda GR18090.

algorítmico IACEI propuesto, cuya validación experimental se discute en la Sección IV. La Sección V presenta finalmente las conclusiones de este estudio, así como líneas de trabajo futuro.

II. Formulación del problema

Los análisis filogenéticos examinan datos biológicos de distintos organismos para definir hipótesis sobre su historia evolutiva [6]. Toman como entrada un alineamiento múltiple de secuencias de dimensiones $N \times M$, donde N es el número de secuencias y M la longitud de secuencia. Los posibles estados que cada carácter de las secuencias puede tomar vienen dados por un alfabeto de estados Λ que codifica, en el caso de proteínas, los 20 posibles aminoácidos junto con sus combinaciones, datos desconocidos y *gaps*. A partir de la comparación de las secuencias es posible inferir un árbol filogenético $T = (V, E)$ que define relaciones ancestro-descendiente. Dentro del conjunto de nodos V, los organismos caracterizados por las secuencias de entrada se etiquetan como nodos hoja, mientras que los nodos internos inferidos representan ancestros hipotéticos. Las relaciones de parentesco entre nodos vienen determinadas por el conjunto de ramas E.

El proceso de inferencia tiene por objeto identificar el árbol filogenético que maximice o minimice un determinado criterio de optimalidad biológico. Puede por tanto ser formulado como un problema de optimización, en el cual el número de posibles soluciones candidatas viene dado por el doble factorial $(2N-5)!!$. Esto implica un crecimiento exponencial del espacio de búsqueda en función de N, siendo representativo el hecho de que, incluso en el caso de instancias pequeñas del problema como $N = 50$, el número de soluciones se aproxima al número de Eddington [10]. La definición del criterio de optimalidad añade una capa adicional de complejidad, al existir distintas tendencias conflictivas a la hora de validar la calidad biológica de una hipótesis filogenética [11]. Estas cuestiones exigen la definición de métodos multiobjetivo robustos que permitan abordar de forma efectiva y eficiente el problema.

Este trabajo afronta la inferencia filogenética usando dos funciones objetivo: parsimonia y verosimilitud. El primer objetivo, parsimonia, mide el número de mutaciones observadas entre especies emparentadas. Se dará prioridad a la definición del árbol filogenético más simple, esto es, aquella solución $T = (V, E)$ que minimice el valor de parsimonia $P(T)$:

$$P(T) = \sum_{i=1}^{M} \sum_{(u,v)\in E} C(u_i, v_i), \tag{1}$$

donde $(u, v) \in E$ hace referencia a la rama entre los nodos $u, v \in V$, u_i y v_i son los valores de estado del carácter i-ésimo de las secuencias correspondientes a u y v, y $C(u_i, v_i)$ indica si se ha producido una mutación o no entre u y v ($C(u_i, v_i)$=1 si $u_i \neq v_i$ y $C(u_i, v_i)$=0 en caso contrario).

Por su parte, la verosimilitud establece una medida probabilística de la plausibilidad de que un árbol filogenético haya dado lugar efectivo a la diversidad observada en las secuencias de entrada. Esta función objetivo emplea en sus cálculos modelos de evolución molecular que definen las probabilidades de mutación de un carácter a otro. En términos de optimización, se dará prioridad a la solución $T = (V, E)$ que maximice el valor de verosimilitud $L(T)$:

$$L(T) = \prod_{i=1}^{M} \sum_{x,y\in\Lambda} \pi_x \left[P_{xy}(t_{ru}) L_p\left(u_i = y\right)\right] \times \left[P_{xy}\left(t_{rv}\right) L_p\left(v_i = y\right)\right], \tag{2}$$

donde Λ es el alfabeto de estados, π_x la probabilidad estacionaria de observar el valor de estado x, $r \in V$ la raíz de T con hijos $u, v \in V$, $P_{xy}(t)$ la probabilidad de mutación desde x a otro estado y en un tiempo t (siendo t_{ru} y t_{rv} la longitud de las ramas $(r, u), (r, v) \in E$), y $L_p(u_i = y)$, $L_p(v_i = y)$ las verosimilitudes parciales de observar y en u_i y v_i.

III. Cooperación Inter-algoritmo con Isla Élite

A fin de resolver el problema de la inferencia filogenética, proponemos una aproximación cooperativa inter-algorítmica basada en la información proporcionada por las tres tendencias de diseño fundamentales en optimización multiobjetivo. Esta sección detalla las principales características de IACEI.

III-A. Algoritmos Base y Adaptación al Problema

La propuesta incluye como componentes de búsqueda base tres algoritmos multiobjetivo basados en población representativos: NSGA-II [7], IBEA [8] y MOEA/D [9]. NSGA-II emplea los conceptos de ordenación rápida no dominada y distancia de *crowding* para determinar la calidad multiobjetivo de las soluciones encontradas. Por su parte, IBEA integra el cálculo de métricas de calidad multiobjetivo (en nuestro caso, hipervolumen) para evaluar las soluciones. Finalmente, MOEA/D aborda el proceso de optimización descomponiendo el problema original en distintos subproblemas, por ejemplo, mediante la aproximación de Tchebycheff.

La adaptación al problema de estos algoritmos se basa en la definición de una representación matricial del individuo. Dadas N secuencias, una solución vendrá codificada por una matriz de distancias Δ de tamaño $N \times N$, donde cada entrada Δ_{xy} representa la distancia evolutiva entre los organismos x e y. Los operadores evolutivos implementados realizan sus cálculos en base a esta representación matricial. Los operadores de selección en NSGA-II e IBEA aplican torneos binarios usando las medidas de calidad correspondientes (ranking y *crowding* en NSGA-II o fitness basado en indicador en IBEA), mientras que MOEA/D aplica selección considerando la vecindad del subproblema tratado o aleatoriamente sobre toda la población. NSGA-II e IBEA implementan un operador de cruce uniforme que intercambia filas de las matrices padre seleccionadas. Por su parte, MOEA/D aplica el operador de evolución diferencial para conformar las entradas de la matriz hija. El operador de mutación es común y consiste en la modificación de entradas, aleatoriamente seleccionadas, de la matriz según la distribución gamma de variación genética observada en las secuencias. Dado que el resultado de estos operadores es matricial, utilizamos el método de *neighbour-joining* BIONJ [6] para generar el árbol filogenético correspondiente.

III-B. Diseño del Modelo Inter-algorítmico IACEI

El modelo inter-algorítmico de IACEI explota la cooperación de NSGA-II, IBEA y MOEA/D para potenciar la calidad de solución obtenida en el proceso de optimización. El diseño implementa estrategias de evolución paralela y migración, que se llevan a cabo de acuerdo al feedback proporcionado por los mecanismos de ranking Pareto y estimación de densidad, indicadores de calidad y descomposición. Adicionalmente, se integra un cuarto componente, denominado isla élite, el cual gobierna la búsqueda gestionando la comunicación de individuos prometedores, así como la identificación y selección del mejor algoritmo. Empleando medidas de rendimiento, la isla élite contribuye a la búsqueda ejecutando una instancia adicional del algoritmo que demuestre mejores resultados dentro de un determinado margen temporal. Esta idea introduce matices de competitividad dentro de la cooperación inter-algorítmica, dado que el algoritmo élite cambiará dinámicamente de acuerdo al éxito de los componentes base a lo largo de la ejecución.

Empleando el estándar Message Passing Interface (MPI) [12], al inicio de IACEI se lanzan 4 procesos, los cuales gestionan la ejecución de NSGA-II, IBEA, MOEA/D y la isla élite. NSGA-II, IBEA y MOEA/D operan sobre subpoblaciones separadas P_{NSGA}, P_{IBEA}, P_{MOEAD} de tamaño *subpopSize*. La isla élite, ilustrada en el Pseudocódigo 1, se mantiene a la espera durante las etapas iniciales de la ejecución. Con una frecuencia de *migrInterval* evaluaciones, los componentes base interactúan con la isla élite comunicando sus subpoblaciones (línea 4 del Pseudocódigo 1). La isla élite lleva a cabo entonces la evaluación comparativa de los algoritmos, calculando el rendimiento de cada uno de ellos (denotado como $Perf_{NSGA}$, $Perf_{IBEA}$, $Perf_{MOEAD}$, línea 5).

Este procedimiento de evaluación se realiza del siguiente modo. Las subpoblaciones comunicadas son combinadas y ordenadas, en primer lugar, usando ranking Pareto y distancias de *crowding*, permitiéndose así la identificación de los mejores *subpopSize* individuos desde la primera perspectiva multiobjetivo implementada. Se inspeccionan los identificadores de los procesos que contribuyeron dentro de esta lista de mejores individuos, incrementando los valores $Perf_{NSGA}$, $Perf_{IBEA}$, $Perf_{MOEAD}$ por cada solución aportada. Este procedimiento se repite desde las perspectivas multiobjetivo restantes, usando a continuación la selección basada en indicadores de calidad y finalmente la descomposición de Tchebycheff. Estos valores de rendimiento son usados por la isla élite para identificar e instanciar dentro de su proceso el algoritmo más satisfactorio. El algoritmo seleccionado operará sobre una subpoblación separada P_{ELITE} que se inicializa en la primera interacción (líneas 7–9) con los *subpopSize* individuos más prometedores del conjunto $P_{NSGA} \cup P_{IBEA} \cup P_{MOEAD}$. Estos individuos prometedores son identificados conforme a la estrategia multiobjetivo implementada en el algoritmo élite seleccionado.

Antes de lanzar la instancia del algoritmo élite, se lleva a cabo la gestión centralizada de las migraciones (línea 11). El procedimiento de migración interactúa con los 3 procesos responsables de ejecutar los algoritmos base, enviando un total de *migrSize* individuos a cada uno. Concretamente, el algoritmo i recibirá los mejores *migrSize/3* individuos de P_{ELITE}, *migrSize/3* individuos de P_j y *migrSize/3* individuos de P_k, donde $i \neq j, k$. La idea subyacente consiste en permitir a cada algoritmo base beneficiarse de migrantes procedentes del resto de estrategias multiobjetivo involucradas en la cooperación inter-algorítmica, así como incorporar soluciones de gran calidad de la isla élite. La política de integración aplica la estrategia de evaluación multiobjetivo implementada en el algoritmo receptor para determinar la inclusión o no de los individuos migrantes en la subpoblación. De este modo, es posible incorporar soluciones prometedoras de los otros componentes dentro del proceso de optimización del algoritmo receptor para potenciar sus capacidades de búsqueda.

Pseudocódigo 1 IACEI: Isla Élite

Entrada: *subpopSize* (tamaño de subpoblación), *migrInterval* (intervalo de migración), *migrSize* (número de individuos migrantes), *NParam* (parámetros del NSGA-II), *IParam* (parámetros del IBEA), *MParam* (parámetros del MOEA/D), *maxEval* (condición de parada, número máximo de evaluaciones).
Salida: *PF* (soluciones no dominadas encontradas).

```
1:  P ← Inicializar Estructuras Poblacionales (subpopSize), PF ← ∅
2:  mientras ! Condición de Parada (maxEval) hacer
3:      /* 1.- Recepción de poblaciones y evaluación multiobjetivo */
4:      P_NSGA, P_IBEA, P_MOEAD ← Recibir Individuos (i, migrInterval, subpopSize) /*
        ∀ i ∈ {NSGA-II, IBEA, MOEA/D} */
5:      Perf_NSGA, Perf_IBEA, Perf_MOEAD ← Calcular Rendimiento Multiobjetivo (P_NSGA,
        P_IBEA, P_MOEAD, subpopSize)
6:      bestMOEA ← Seleccionar Algoritmo Élite (Perf_NSGA, Perf_IBEA, Perf_MOEAD)
7:      si P_ELITE = ∅ entonces
8:          P_ELITE ← Inicializar Población Élite (P_NSGA, P_IBEA, P_MOEAD, subpopSize,
            bestMOEA)
9:      fin si
10:     /* 2.- Migración de individuos hacia NSGA-II, IBEA y MOEA/D */
11:     Enviar Migrantes (i, P_p, migrSize) /* ∀ i ∈ {NSGA-II, IBEA, MOEA/D}, ∀
        P_p ∈ P : p ≠ i */
12:     /* 3.- Ejecutar instancia del mejor algoritmo sobre la población élite */
13:     Seleccionar (bestMOEA)
14:         Caso NSGA-II: P_ELITE, PF ← Ejecutar NSGA-II (NParam, migrInterval)
15:         Caso IBEA: P_ELITE, PF ← Ejecutar IBEA (IParam, migrInterval)
16:         Caso MOEA/D: P_ELITE, PF ← Ejecutar MOEA/D (MParam, migrInterval)
17: fin mientras
18: PF ← Componer Frente Pareto (PF, PF_i) /*∀ i ∈ {NSGA-II, IBEA, MOEA/D}*/
```

Tras completar las migraciones, los 4 procesos continúan con los pasos evolutivos, ejecutando el correspondiente a la isla élite el algoritmo seleccionado (líneas 13-16) hasta la siguiente interacción entre procesos. P_{ELITE} evolucionará de forma separada sin recibir ningún otro feedback del resto de procesos salvo el algoritmo a ser ejecutado. Esto conlleva la definición de dos líneas de cooperación en IACEI: 1) la contribución del método de búsqueda, desde los algoritmos base a la isla élite, y 2) la contribución de individuos prometedores, desde la gestión élite a los algoritmos base. Satisfecha la condición de parada, se compondrá el frente de Pareto conjunto y será devuelto como salida del método (línea 18).

IV. Resultados Experimentales

La evaluación experimental de la propuesta se ha realizado sobre una infraestructura hardware de 48 cores con procesadores AMD Opteron 6174 a 2,2GHz, 12MB de caché L3 y 64GB de RAM DDR3. En el aspecto software, se ha utilizado el sistema operativo Ubuntu 18.04LTS, el compilador GCC 5.2.1 y la distribución MPI MPICH3. Las instancias del problema consideradas incluyen cinco conjuntos reales de proteínas:

Tabla I
COMPARATIVAS CON ALGORITMOS BASE: HIPERVOLUMEN I_H Y DISTANCIA GENERACIONAL INVERTIDA IGD

Hipervolumen I_H				
Instancia	IACEI	NSGA-II	IBEA	MOEA/D
M67x11333	**83,52 %±0,19**	78,79 %±0,33	75,85 %±0,39	80,52 %±0,60
M88x3329	**60,41 %±0,24**	55,98 %±0,54	57,42 %±0,56	58,08 %±0,55
M187x814	**65,45 %±0,58**	58,09 %±1,82	59,38 %±2,21	61,82 %±1,50
M260x1781	**76,20 %±0,36**	65,15 %±1,32	71,32 %±0,78	73,43 %±0,50
M355x1263	**75,93 %±0,69**	71,77 %±0,76	72,45 %±0,74	73,47 %±1,00
Distancia generacional invertida IGD				
M67x11333	**1,21 %**	5,24 %	8,18 %	3,28 %
M88x3329	**4,86 %**	13,64 %	10,71 %	8,83 %
M187x814	**3,06 %**	10,44 %	9,79 %	6,49 %
M260x1781	**3,18 %**	14,34 %	6,56 %	5,82 %
M355x1263	**2,49 %**	6,90 %	6,34 %	5,65 %

Tabla II
COMPARATIVAS CON ALGORITMOS BASE: RELACIÓN DE COBERTURA SC

Relación de cobertura SC(IACEI, Base)			
Instancia	NSGA-II	IBEA	MOEA/D
M67x11333	**95,28 %**	**94,64 %**	**92,05 %**
M88x3329	**100,00 %**	**92,86 %**	**100,00 %**
M187x814	**88,52 %**	**59,09 %**	**83,78 %**
M260x1781	**79,75 %**	**72,53 %**	**93,50 %**
M355x1263	**90,38 %**	**79,49 %**	**76,29 %**
Relación de cobertura SC(Base, IACEI)			
M67x11333	1,89 %	1,89 %	3,77 %
M88x3329	0,00 %	6,67 %	0,00 %
M187x814	7,41 %	14,81 %	3,70 %
M260x1781	11,30 %	14,78 %	2,61 %
M355x1263	12,82 %	12,82 %	15,38 %

- M67x11333: 67 secuencias de 11333 caracteres correspondientes a proteínas de eubacterias [13].
- M88x3329: 88 secuencias de 3329 caracteres correspondientes a proteínas de hongos termófilos [14].
- M187x814: 187 secuencias de 814 caracteres de proteínas de hongos micorrizos [15].
- M260x1781: 260 secuencias de 1781 caracteres de proteínas de la remolacha (*Beta vulgaris*) [16].
- M355x1263: 355 secuencias de 1263 caracteres correspondientes a proteínas de levaduras [17].

En cada experimento se realizaron 31 ejecuciones independientes de cada método evaluado por instancia. Tres métricas multiobjetivo han sido utilizadas para validar la contribución de la propuesta con respecto a sus componentes base y otros modelos cooperativos alternativos. Consideramos las métricas de 1) *hipervolumen* (I_H), para medir el área del espacio objetivo dominado por las soluciones generadas; 2) *distancia generacional invertida* (*IGD*), calculada con respecto a un frente de Pareto de referencia producto de la combinación de los resultados de todos los métodos; y 3) *relación de cobertura* (*SC*), para medir la fracción de soluciones que cada algoritmo cubre (por dominancia Pareto débil) sobre el resto y viceversa.

Se realizaron estudios paramétricos para determinar la configuración óptima de los algoritmos evaluados. De acuerdo a dichos estudios, los valores de los parámetros de entrada fueron establecidos como sigue:

- Parámetros comunes: probabilidad de cruce = 70 %, probabilidad de mutación = 5 %, máximo número de evaluaciones = 12000.
- Parámetros específicos de IBEA: escalado kappa k = 0,05, punto de referencia de hipervolumen = (2,2).
- Parámetros específicos de MOEA/D: parámetros de control de evolución diferencial F = 0,5 y CR = 75 %, tamaño de vecindad de subproblema = 4, probabilidad de selección de padres en la vecindad = 25 % y máximo número de soluciones a actualizar = 1.
- Parámetros cooperativos de IACEI: tamaño de subpoblación *subpopSize* = 48 individuos, intervalo de migración *migrInterval* = 750 evaluaciones y número de migrantes *migrSize* = *subpopSize*/2.

IV-A. Comparativas con Componentes Individuales

Abordaremos primero la evaluación de IACEI con respecto a las versiones no cooperativas de sus componentes base: NSGA-II, IBEA y MOEA/D. La Tabla I proporciona los hipervolúmenes medianos de cada método (considerando el mismo número de evaluaciones), junto con el rango intercuartílico. Esta tabla también recoge los resultados para la distancia generacional invertida. Los resultados de la relación de cobertura son presentados en la Tabla II.

Centrándonos en el hipervolumen, se puede observar que la propuesta inter-algorítmica IACEI consigue ir un paso más allá en términos de calidad multiobjetivo con respecto a NSGA-II, IBEA y MOEA/D en el problema abordado. Los hipervolúmenes medianos de IACEI se sitúan dentro del rango 60,4 % (M88x3329) a 83,5 % (M67x11333), dando lugar a un resultado medio del 72,3 % al considerar todas las instancias. En comparación, NSGA-II obtuvo un valor de hipervolumen medio del 65,9 %, IBEA del 67,2 % y MOEA/D del 69,4 %.

En el caso de la distancia generacional invertida, la Tabla I da cuenta de la obtención por parte de IACEI de aproximaciones mejores al frente de Pareto de referencia. Mientras que los algoritmos base obtienen distancias de hasta el 14,3 % (NSGA-II), 10,7 % (IBEA) y 8,8 % (MOEA/D), IACEI se mantiene estable en valores por debajo del 5 % en todas las instancias analizadas. Para ilustrar estas diferencias, se presenta en la Figura 1 una comparativa entre los frentes medianos de cada algoritmo en M67x11333. Puede observarse que el frente de IACEI presenta unas propiedades de convergencia y forma más satisfactorias que las obtenidas por los algoritmos base.

Los resultados mostrados en la Tabla II para la relación de cobertura también sugieren la mejora que supone la propuesta. Las comparativas indican que IACEI es capaz de cubrir porcentajes de soluciones del 79,8 %–100,0 % sobre NSGA-II, 59,1 %–94,6 % sobre IBEA y 76,3 %–100,0 % sobre MOEA/D. Por su parte, la fracción de soluciones de IACEI que son cubiertas por los algoritmos base es, como mucho, del 15,4 %. Estas tres métricas por tanto apuntan, desde distintas perspectivas, a la mejora en calidad multiobjetivo que supone el empleo de las estrategias de cooperación inter-algorítmica implementadas en IACEI en comparación a las versiones individuales no cooperativas de NSGA-II, IBEA y MOEA/D.

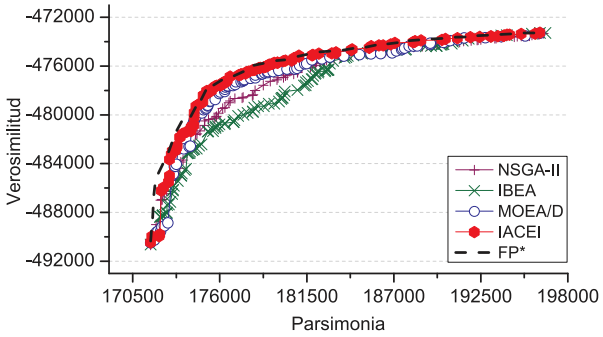


Figura 1. Comparativa de frentes de Pareto de hipervolumen mediano para la instancia M67x11333, con respecto al frente óptimo (FP*)

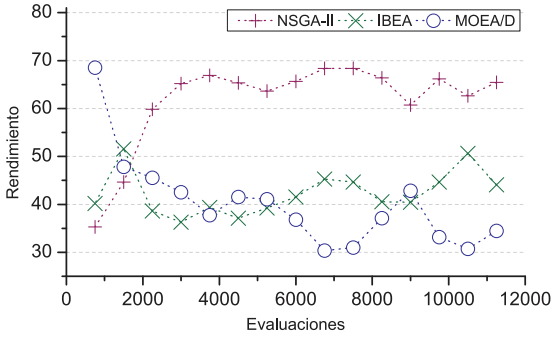


Figura 2. Evolución del rendimiento medio de cada tendencia algorítmica durante la ejecución de IACEI, sobre la instancia M88x3329

Tabla III
COMPARATIVAS CON MODELO DE ISLAS ESTÁNDAR (INTERCONEXIÓN COMPLETA *Full* Y ANILLO *Ring*): I_H E IGD

Hipervolumen I_H			
Instancia	IACEI	*Full*	*Ring*
M67x11333	**83,52 %±0,19**	82,59 %±0,17	82,30 %±0,26
M88x3329	**60,41 %±0,24**	60,17 %±0,27	59,93 %±0,22
M187x814	**65,45 %±0,58**	63,38 %±0,98	62,19 %±1,31
M260x1781	**76,20 %±0,36**	74,81 %±0,62	74,02 %±0,73
M355x1263	**75,93 %±0,69**	74,12 %±0,64	73,53 %±0,65
Distancia generacional invertida IGD			
M67x11333	**1,21 %**	1,51 %	1,86 %
M88x3329	**4,86 %**	5,50 %	5,98 %
M187x814	**3,06 %**	4,82 %	6,11 %
M260x1781	**3,18 %**	4,49 %	5,31 %
M355x1263	**2,49 %**	4,85 %	5,46 %

Tabla IV
COMPARATIVAS CON MODELO DE ISLAS ESTÁNDAR (INTERCONEXIÓN COMPLETA *Full* Y ANILLO *Ring*): SC

Relación de cobertura SC				
Instancia	SC(IACEI, *Full*)	SC(*Full*, IACEI)	SC(IACEI, *Ring*)	SC(*Ring*, IACEI)
M67x11333	**50,55 %**	33,96 %	**66,67 %**	15,09 %
M88x3329	**64,29 %**	40,00 %	**60,00 %**	33,33 %
M187x814	**69,39 %**	16,67 %	**65,79 %**	14,82 %
M260x1781	**72,18 %**	20,87 %	**87,74 %**	13,91 %
M355x1263	**76,54 %**	17,95 %	**79,49 %**	11,54 %

IV-B. Comparativas con Otros Modelos Cooperativos

En segundo lugar, introducimos comparativas entre IACEI y otros modelos cooperativos alternativos. Concretamente, consideraremos el modelo de islas estándar bajo dos configuraciones topológicas diferentes: 1) topología de interconexión completa y 2) topología de anillo. Ambas implementaciones del modelo estándar incluyen también como componentes base a NSGA-II, IBEA y MOEA/D y el mismo criterio de parada.

Las Tablas III y IV presentan la comparativa de rendimiento multiobjetivo entre IACEI y las configuraciones del modelo estándar. En términos generales, los resultados multiobjetivo obtenidos apuntan a mejoras al emplear las estrategias definidas en la propuesta IACEI. Los resultados de hipervolumen de la Tabla III denotan que las diferencias entre IACEI y las alternativas del modelo estándar se vuelven más notables al considerar instancias con un mayor número de secuencias (concretamente, en M187x814, M260x1781 y M355x1263). Habida cuenta de que el número de secuencias gobierna el tamaño del espacio de búsqueda, esta apreciación sugiere que la propuesta dar lugar a un procesamiento más satisfactorio de conjuntos de datos complejos.

En lo que respecta a la distancia generacional invertida (Tabla III), la diferencia más notable con respecto a la alternativa de interconexión completa se da en la instancia con mayor número de secuencias, M355x1263. La comparativa con la alternativa de anillo también sugiere el rendimiento notable de IACEI en instancias complejas del problema, conforme a las distancias obtenidas en M187x814, M260x1781 y M355x1263. Es en estas dos últimas instancias donde se verifican los porcentajes más notables de cobertura de IACEI sobre el modelo estándar, dominando fracciones de soluciones por encima del 70 % (Tabla IV). En cambio, los porcentajes cubiertos por el modelo estándar se limitan, en término medio, al 25,9 % (interconexión completa) y 17,7 % (anillo).

Estos resultados de rendimiento multiobjetivo apuntan a la bondad de las estrategias inter-algorítmicas implementadas en IACEI. En este sentido, merece la pena destacar las capacidades dinámicas de búsqueda que esta aproximación permite de acuerdo a las características de la instancia analizada y el estado del proceso de optimización en un determinado instante de tiempo. Como ejemplo, la Figura 2 representa la evolución del rendimiento conseguido por cada componente base a lo largo de la ejecución de IACEI para la instancia M88x3329.

IV-C. Comparativas Biológicas

Como último punto de esta evaluación, estudiaremos la calidad filogenética de los árboles de IACEI con respecto a otras herramientas biológicas de la literatura. Consideraremos, por un lado, dos métodos basados en parsimonia, TNT [18] y ProtPars [19]. Por otra parte, las comparativas de verosimilitud se realizarán usando como referencia los métodos RAxML [20] y GARLI [21]. Para llevar a cabo esta comparativa, se examinó la calidad de los puntos extremos generados por IACEI en las ejecuciones de hipervolumen mediano, así como los resultados medianos producidos por cada método biológico tras 31 ejecuciones independientes por experimento.

La Tabla V presenta los resultados de parsimonia y verosimilitud observados por instancia y método. Desde la

Tabla V
COMPARATIVAS DE CALIDAD FILOGENÉTICA CON RESPECTO A HERRAMIENTAS BIOLÓGICAS

Parsimonia $P(T)$			
Instancia	IACEI	TNT	ProtPars
M67x11333	**171623**	**171623**	173765
M88x3329	**33490**	**33490**	33944
M187x814	**29847**	**29847**	29955
M260x1781	**43529**	**43529**	44495
M355x1263	**54823**	**54823**	55328
Verosimilitud $L(T)$			
Instancia	IACEI	RAxML	GARLI
M67x11333	**-473260,21**	**-473260,21**	-473264,64
M88x3329	**-149094,84**	**-149094,84**	-149110,96
M187x814	**-133875,38**	-133877,79	-133875,41
M260x1781	**-163895,46**	-163912,21	-163981,71
M355x1263	**-231184,74**	-231193,88	-231293,81

perspectiva de parsimonia, IACEI es capaz de alcanzar la calidad de resultados de la herramienta de referencia TNT, mejorando ambas aproximaciones a ProtPars en todas las instancias analizadas. La calidad filogenética de los resultados obtenidos es confirmada en la comparativa de verosimilitud, donde IACEI consigue mejorar los resultados obtenidos tanto por GARLI como RAxML en las instancias con mayor número de secuencias (M187x814, M260x1781 y M355x1263). Por tanto, este análisis comparativo da cuenta del interés de la propuesta para abordar problemas de optimización complejos como el estudiado en este trabajo.

V. CONCLUSIONES

Este trabajo ha explorado la cooperación inter-algorítmica para afrontar el problema de la inferencia filogenética. Se ha definido una propuesta que integra a algoritmos representativos de las tres tendencias de diseño fundamentales en optimización multiobjetivo: NSGA-II, IBEA y MOEA/D. La propuesta, denominada IACEI, extiende los mecanismos de evolución paralela y migración mediante la introducción de un elemento de control denominado isla élite. Esta isla élite implementa estrategias dinámicas basadas en medidas de rendimiento para 1) lanzar instancias adicionales de los algoritmos en función de su éxito en cada etapa de la ejecución y 2) definir distribuciones equilibradas de migrantes prometedores conforme a las tres tendencias multiobjetivo consideradas.

La evaluación experimental de la propuesta se ha llevado a cabo sobre cinco instancias reales del problema, analizando la calidad multiobjetivo con respecto a los componentes individuales y otros modelos inter-algorítmicos. Los resultados obtenidos muestran que IACEI consigue abordar de forma satisfactoria escenarios complejos de optimización, de acuerdo a las mejoras observadas en los conjuntos de datos con mayor número de secuencias. Estas mejoras también tienen implicaciones desde una perspectiva biológica, conforme apuntan las comparativas de calidad filogenética con respecto a otras herramientas comúnmente adoptadas en la literatura.

En cuanto a trabajo futuro, se investigará en primer lugar la integración de esquemas paralelos intra-algorítmicos dentro de IACEI para explotar las capacidades de cómputo paralelo de las plataformas heterogéneas CPU+GPU actuales. La introducción de mecanismos de optimización adaptativos adicionales, por ejemplo *resource allocation*, también representa una línea prometedora para focalizar los esfuerzos de optimización sobre las regiones más difíciles del frente de Pareto. Finalmente, abordaremos el estudio de IACEI en escenarios de optimización *many*-objetivo (NSGA-III, RVEA, etc.).

REFERENCIAS

[1] T. K. Attwood, S. R. Pettifer, and D. Thorne, *Bioinformatics Challenges at the Interface of Biology and Computer Science: Mind the Gap*. Oxford, UK: Wiley Blackwell, 2016.

[2] D. Kimovski, J. Ortega, A. Ortiz, and R. Baños, "Parallel alternatives for evolutionary multi-objective optimization in unsupervised feature selection," *Expert Syst. Appl.*, vol. 42, no. 9, pp. 4239–4252, 2015.

[3] M. Kurdi, "An improved island model memetic algorithm with a new cooperation phase for multi-objective job shop scheduling problem," *Computers & Industrial Engineering*, vol. 111, pp. 183–201, 2017.

[4] J. Ortega, D. Kimovski, J. Q. Gan, A. Ortiz, and M. Damas, "A Parallel Island Approach to Multiobjective Feature Selection for Brain-Computer Interfaces," in *IWANN 2017: Advances in Computational Intelligence*, ser. LNCS, vol. 10305. Springer Verlag, 2017, pp. 16–27.

[5] Y.-J. Gong, W.-N. Chen, Z.-H. Zhan, J. Zhang, Y. Li, Q. Zhang, and J.-J. Li, "Distributed evolutionary algorithms and their models: A survey of the state-of-the-art," *Appl. Soft Comput.*, vol. 34, pp. 286–300, 2015.

[6] M. Steel, *Phylogeny: Discrete and Random Processes in Evolution*. Philadelphia: Society for Industrial and Applied Mathematics, 2016.

[7] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A Fast and Elitist Multi–Objective Genetic Algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, 2002.

[8] E. Zitzler and S. Künzli, "Indicator-Based Selection in Multiobjective Search," in *Parallel Problem Solving From Nature VIII*, ser. LNCS, vol. 3242. Springer Verlag, 2004, pp. 832–842.

[9] H. Li and Q. Zhang, "Multiobjective Optimization Problems With Complicated Pareto Sets, MOEA/D and NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 13, no. 2, pp. 284–302, 2009.

[10] A. Rokas, "Phylogenetic Analysis of Protein Sequence Data Using the Randomized Axelerated Maximum Likelihood (RAxML) Program," *Current Protocols in Molecular Biology*, vol. 96, pp. 19.11.1–14, 2011.

[11] S. Santander-Jiménez and M. A. Vega-Rodríguez, "Applying a Multiobjective Metaheuristic Inspired by Honey Bees to Phylogenetic Inference," *BioSystems*, vol. 114, no. 1, pp. 39–55, 2013.

[12] W. Gropp, W. Lusk, and A. Skjellum, *Using MPI: Portable Parallel Programming with the Message Passing Interface. 3rd edition*. Cambridge, MA, USA: The MIT Press, 2014.

[13] D. He, O. Fiz-Palacios, C. Fu, J. Fehling, C. Tsai, and S. L. Baldauf, "An Alternative Root for the Eukaryote Tree of Life," *Current Biology*, vol. 24, no. 4, pp. 465–470, 2014.

[14] I. Morgenstern *et al.*, "A molecular phylogeny of thermophilic fungi," *Fungal Biology*, vol. 116, no. 4, pp. 489–502, 2012.

[15] A. Kovalchuk, A. Kohler, F. Martin, and F. O. Asiegbu, "Diversity and evolution of ABC proteins in mycorrhiza-forming fungi," *BMC Evolutionary Biology*, vol. 15, no. 249, pp. 1–19, 2015.

[16] R. Stracke, D. Holtgräwe, J. Schneider, B. Pucker, T. R. Sörensen, and B. Weisshaar, "Genome-wide identification and characterisation of R2R3-MYB genes in sugar beet (Beta vulgaris)," *BMC Plant Biology*, vol. 14, no. 249, pp. 1–17, 2014.

[17] P. J. Dias and I. Sá-Correia, "The drug:H+ antiporters of family 2 (DHA2), siderophore transporters (ARN) and glutathione:h+antiporters (GEX) have a common evolutionary origin in hemiascomycete yeasts," *BMC Genomics*, vol. 14, no. 901, pp. 1–22, 2013.

[18] P. A. Goloboff and S. A. Catalano, "TNT version 1.5, including a full implementation of phylogenetic morphometrics," *Cladistics*, vol. 32, no. 3, pp. 221–238, 2016.

[19] J. Felsenstein, "PHYLIP (phylogeny inference package)," http://evolution.genetics.washington.edu/phylip.html, 2014.

[20] A. Stamatakis, "RAxML Version 8: A Tool for Phylogenetic Analysis and Post-Analysis of Large Phylogenies," *Bioinformatics*, vol. 30, no. 9, pp. 1312–1313, 2014.

[21] A. L. Bazinet, D. J. Zwickl, and M. P. Cummings, "A Gateway for Phylogenetic Analysis Powered by Grid Computing Featuring GARLI 2.0," *Systematic Biology*, vol. 63, no. 5, pp. 812–818, 2014.

El efecto de la multiconectividad en el problema del apagado selectivo de redes 5G ultradensas

Alejandro Quiñones-García
Dpto. de LCC
Universidad de Málaga
Málaga
aquiga@lcc.uma.es

Jesús Galeano-Brajones
Dpto. DISIT
Centro Universitario de Mérida
Universidad de Extremadura
Mérida
jgaleanobra@unex.es

Pablo H. Zapata-Cano
Dpto. de TSTC
Universidad de Granada
Granada
pablozapata@correo.ugr.es

Francisco Luna-Valero
Dpto. de LCC
ITIS Software
Universidad de Málaga
Málaga
flv@lcc.uma.es

Javier Carmona-Murillo
Dpto. DISIT
Centro Universitario de Mérida
Universidad de Extremadura
Mérida
jcarmur@unex.es

Juan F. Valenzuela-Valdés
Dpto. de TSTC
Universidad de Granada
Granada
juanvalenzuela@ugr.es

***Resumen*—El despliegue masivo de estaciones base es uno de los pilares básicos para que las nuevas generaciones de redes móviles satisfagan las elevadas demandas de muchas aplicaciones actuales. Esta densificación conlleva, no obstante, un elevado consumo energético que ha de ser minimizado para garantizar la sostenibilidad del sistema en un claro contexto de reducción del impacto ambiental. Se trata, por tanto, de un problema de optimización multiobjetivo en el que no sólo se tienen en cuenta criterios de eficiencia energética, si no también de calidad del servicio. En el contexto de este problema, el objetivo de este trabajo es evaluar cómo la incorporación de una tecnología novedosa, la multiconectividad, que permite a un usuario recibir información de más de una estación base, afecta a la complejidad del espacio de búsqueda subyacente. Para ello, usando diferentes metaheurísticas multiobjetivo, se ha diseñado un estudio experimental con instancias de tamaño y densificación crecientes que ha mostrado que, a pesar de las ventajas que aporta esta tecnología, las aproximaciones al frente de Pareto obtenidas en diferentes escenarios tienen peores indicadores de calidad, i.e., se tiene un espacio de búsqueda más complejo.**

***Index Terms*—Multiconectividad, 5G, Metaheurísticas multiobjetivo**

I. Introducción

El incremento constante en la demanda de servicios con mayores requisitos de ancho de banda (vídeo 4K, realidad aumentada, etc.), menor latencia (e.g., conducción autónoma) o escenarios con una densidad muy elevada de dispositivos conectados, como el Internet de las Cosas, ha motivado el continuo desarrollo de las redes móviles, que ahora se encuentra en los despliegues iniciales de lo que se conoce como la quinta generación (5G), y se investiga ya en la sexta (6G) [1]. De entre los diferentes pilares en los que se sustentan ambas tecnologías [2], la densificación de la red, o despliegue masivo de un gran número de estaciones base de pequeño tamaño (SBSs, por su siglas en Inglés) por km^2, está reconocida como un paradigma fundamental que permite una mayor reutilización del espectro electromagnético y aumenta la capacidad de red. Éstas son las denominadas redes ultradensas (UDNs) [3]. La densificación, no obstante, lleva asociado un incremento del consumo energético del sistema, especialmente en periodos de baja demanda, en los que las SBSs están encendidas, pero sin dar servicio a ningún usuario (e.g., edificios de oficinas que se vacían durante la noche). Para tratar de paliar este problema, una estrategia estandarizada [4] consiste en el encendido/apagado selectivo de SBSs para minimizar la energía consumida, pero tratando simultáneamente de maximizar el nivel de servicio para la demanda existente. Es un problema de optimización combinatoria multiobjetivo que se ha demostrado como NP-completo [5] y cuya resolución se ha planteado en la literatura con el uso de metaheurísticas multiobjetivo [6], [7].

Por otra parte, las UDNs se consideran heterogéneas [8] al contener SBSs de distinta naturaleza en términos de potencia emitida, tamaño y frecuencia de operación, debido a la coexistencia de diferentes tecnologías de radio y, por tanto, los usuarios de la red (UEs o *User Equipments*) pueden estar conectados a varios canales con frecuencias portadoras distintas al mismo tiempo, lo que permite incrementar la tasa de transmisión y/o la fiabilidad [9]. El objetivo principal de este trabajo consiste en estudiar el efecto de la multiconectividad (MC), por una parte, en los objetivos de consumo y capacidad y, por otra, en la búsqueda realizada por diversas metaheurísticas multiobjetivo del dominio. Para ello, usando el mismo

Este trabajo ha sido financiado parcialmente por el Ministerio de Ciencia Innovación y Universidades con los proyectos TIN2016-75097-P, RTI2018-102002-A-I00 y EQC2018-004571-P, por la Consejería de Economía, Conocimiento, Empresas y Universidad de la Junta de Andalucía con los proyectos B-TIC-402-UGR18 y P18.RT.4830, y por la Consejería de Economía e Infraestructuras de la Junta de Extremadura con el proyecto IB18003.

conjunto de instancias de UDNs con diferentes niveles de densificación, tanto de UEs como de SBSs, se han planificado experimentos sin MC (1 portadora), y con MC (con 2 y 3 portadoras) y se han evaluado cuatro algoritmos: dos basados en dominancia de Pareto, y ya usados en trabajos previos, NSGA-II [10] y MOCell [11], y otros dos, MOEA/D [12] y SMS-EMOA [13], basados, respectivamente, en la descomposición de objetivos y búsqueda guiada por indicadores de calidad. Ni MOEA/D ni SMS-EMOA han sido utilizados previamente en el contexto de este problema. Los resultados han mostrado que, por una parte, la MC permite aumentar la capacidad de la red, pero también incrementa la complejidad del problema de optimización subyacente dado que aumenta la relación epistática entre las variables de decisión (e.g., la activación de una BS no sólo permite habilitar nuevas portadoras, si no que también aumentan las interferencias que reducen la capacidad del enlace inalámbrico). Esto último se refleja en que las aproximaciones al frente de Pareto en escenarios con 2 o 3 portadoras obtienen, en muchos casos, peores indicadores de calidad que aquellos en los que la MC no está habilitada.

El trabajo se ha estructurado como sigue. En la siguiente sección se ha formulado el problema de optimización abordado. Las metaheurísticas multiobjetivo utilizadas se describen brevemente en la Sección III. La Sección IV detalla la metodología experimental y el análisis pormenorizado de los resultados obtenidos. Finalmente, la última sección incluye las principales conclusiones alcanzadas, así como las líneas de trabajo futuro que quedan abiertas.

II. Definición del problema

Esta sección detalla el modelado del sistema, el cálculo de los objetivos y cómo se implementa la MC.

II-A. Modelado del sistema

Este proyecto ha considerado un área de servicio metropolitana de 500×500 metros cuadrados, que se ha discretizado usando un grid de 100×100 puntos (también denominados pixeles o elementos del área). Cada uno de ellos cubre un área de 25 m^2 donde la potencia de la señal se asume que es constante. Además, se ha tenido en cuenta la densificación vertical considerando 3 elementos de área verticales, correspondientes a las diferentes alturas de las BTS. Se han definido 10 regiones diferentes con distintas condiciones de propagación, de forma que, para calcular la potencia recibida en cada punto, $P_{rx}[dBm]$, se ha utilizado el siguiente modelo [6] [7]:

$$P_{rx}[dBm] = P_{tx}[dBm] + PLoss[dB] \quad (1)$$

donde, P_{rx} es la potencia recibida [dBm], P_{tx} es la potencia transmitida [dBm], y $PLoss$ son las pérdidas totales, que dependen de la región de propagación dada, y se calculan como:

$$PLoss[dB] = GA + PA \quad (2)$$

donde, GA es la ganancia total de ambas antenas, y PA son las pérdidas de propagación en espacio libre, calculadas como:

$$PA[dB] = \left(\frac{\lambda}{2 \cdot \pi \cdot d}\right)^K \quad (3)$$

Tabla I
Parámetros de configuración de los escenarios de UDN

Celdas		Eq.	LL	LM	LH	ML	MM	MH	HL	HM	HH
Micro	G_{tx}	(2)	12								
	f	(5)	5 GHz (BW = 500 MHz)								
	α	(8)	15								
	β	(8)	10000								
	δ	(8)	1								
	$\rho[W]$	(8)	1								
	n_{tx}		8								
	n_{rx}		2								
	λ_P^{micro}		100	100	100	200	200	200	300	300	300
Pico	G_{tx}	(2)	20								
	f	(5)	20 GHz (BW = 2000 MHz)								
	α	(8)	9								
	β	(8)	6800								
	δ	(8)	0.5								
	$\rho[W]$	(8)	1								
	n_{tx}		64								
	n_{rx}		4								
	λ_P^{pico}		500	500	500	600	600	600	700	700	700
Femto	G_{tx}	(2)	28								
	f	(5)	68 GHz (BW = 6800 MHz)								
	α	(8)	5.5								
	β	(8)	4800								
	δ	(8)	0.2								
	$\rho[W]$	(8)	1								
	n_{tx}		256								
	n_{rx}		8								
	λ_P^{femto}		1000	1000	1000	2000	2000	2000	3000	3000	3000
UEs	λ_P^{UE}		1000	2000	3000	1000	2000	3000	1000	2000	3000

donde d es la distancia euclídea a la SBS, K es el exponente de pérdidas, que varía aleatoriamente entre $[2.0, 4.0]$ para cada una de las 10 regiones diferentes. La relación señal/interferencia más ruido (SINR) para UE k, se calcula como:

$$SINR_k = \frac{P_{rx,j,k}[mW]}{\sum_{i=1}^{M} P_{rx,i,k}[mW] - P_{rx,j,k}[mW] + P_n[mW]} \quad (4)$$

donde $P_{rx,j,k}$ es la potencia recibida por el UE k desde la celda j, el sumatorio es la potencia total recibida por el UE k desde todas las celdas que operan en la misma frecuencia que j, y P_n es la potencia de ruido, calculada como:

$$P_n[dBm] = -174 + 10 \cdot \log_{10} BW_j \quad (5)$$

siendo BW_j el ancho de banda del SBS j, definido como el 10 % de la frecuencia de funcionamiento del SBS (ver Tabla I).

Por último, la capacidad del UE se ha calculado según la tecnología MIMO (Multiple-Input Multiple-Output), considerando múltiples antenas de transmisión y recepción, representada en [14]. Así, suponemos que la potencia de transmisión de cada antena es P_{tx}/n_tx, donde n_tx indica el número de antenas de transmisión. Entonces, si consideramos que los subcanales están desacoplados, sus capacidades pueden sumarse, y la capacidad global del canal del equipo de usuario k puede estimarse mediante la fórmula de capacidad de Shannon:

$$C_k^j[bps] = BW_k^j[Hz] \cdot \sum_{i=1}^{r} \log_2\left(1 + \frac{SINR_k \cdot \lambda_i}{n_{tx}}\right) \quad (6)$$

donde $\sqrt{\lambda_i}$ es el valor singular de la matriz del canal $\mathbf{H}$, de dimensiones $n_{rx} \times n_{tx}$ (es decir, antenas recepción y antenas de transmisión). Obsérvese que tanto n_{rx} como n_{tx} dependen

del tipo de celda (ver Tabla I). BW_k^j es el ancho de banda asignado al UE k cuando se conecta a la celda j, suponiendo una programación round robin, esto es:

$$BW_k^j = \frac{BW_j}{N_j} \tag{7}$$

donde N_j es el número de UEs conectados a la celda j, los cuales se conectan a la celda que proporciona la mayor SINR, independientemente de su tipo.

Para construir una red heterogénea, se consideran tres tipos diferentes de celdas de tamaño creciente y frecuencia decreciente: femtoceldas, picoceldas y macroceldas. Cabe destacar que el número de antenas de transmisión de cada tipo de celda aumenta con la frecuencia, pasando de 8 antenas de transmisión (microcelda) a 256 (femtocelda), así como la directividad de las antenas que componen el emisor (la directividad aumenta con la frecuencia y disminuye con el tamaño, de ahí la necesidad de usar más antenas para frecuencias más altas). Del mismo modo, suponemos que los UEs de "alta capacidad", que se conectarán preferentemente a celdas pequeñas (picoceldas y femtoceldas), implementarán un mayor número de antenas de recepción (4 para picoceldas y 8 para femtoceldas). Tanto las celdas como los UEs se despliegan utilizando Procesos de Poisson (PPP) independientes con diferentes densidades, definidas por λ_P^{Cells} y λ_P^{UE}, respectivamente.

El consumo de energía de un transmisor se calcula en base al modelo presentado en [15], que considera que el dispositivo está transmitiendo a través de la fibra backhauling. Por lo tanto, el consumo de energía regular de SBS j, P_j, se expresa como:

$$P_j = \alpha \cdot P + \beta + \delta \cdot S + \rho \tag{8}$$

donde P denota la potencia transmitida o radiada del transmisor, el coeficiente α representa la eficiencia de la potencia de transmisión producida por un amplificador de radiofrecuencia y las pérdidas del alimentador, la potencia disipada debido al procesamiento de la señal y la refrigeración del emplazamiento se denota por β y el consumo de potencia dinámico por unidad de datos viene dado por δ, siendo S la demanda de tráfico real proporcionada por la celda servidora. Por último, el consumo de energía del dispositivo transmisor está representado por el coeficiente ρ. Sin embargo, para considerar un modelo de consumo de energía preciso, también debe tenerse en cuenta la energía consumida por el aire acondicionado y la alimentación de la estación base [16]. Esto se ha denominado potencia de mantenimiento y se fija en 2W/SBS para cualquier SBS que contenga al menos una celda activa.

La parametrización detallada de los escenarios abordados se incluye en la Tabla I, en la que la columna Eq. vincula el parámetro con la ecuación correspondiente en la formulación detallada anteriormente. Los nombres de las nueve últimas columnas, XY, representan las densidades de despliegue de SBS y UE. Los niveles de despliegue, para baja, media o alta densidad, respectivamente, se denotan de modo {L, M, H}, y se configuran con los parámetro de la PPP λ_P^{Cell} para las celdas y λ_P^{UE} para los UEs. Los parámetros G_{tx} y f de cada tipo de celda se refieren a la ganancia de transmisión y a la frecuencia de funcionamiento (y su ancho de banda disponible) de la antena, respectivamente, siendo n_{tx} y n_{rx} el número de antenas de transmisión y recepción. Por último, también se incluyen los parámetros del modelo de consumo de energía descrito anteriormente. Por tanto, en este trabajo se han utilizado nueve instancias para evaluar el rendimiento de las diferentes metaheurísticas y operadores específicos.

II-B. Consumo y capacidad en redes 5G

Sea $\mathcal{B}$ el conjunto de los SBS desplegados aleatoriamente. Una solución al problema CSO es una cadena binaria $s \in \{0,1\}^{|\mathcal{B}|}$, donde s_i indica si el SBS i está activado o no. Por lo tanto, el primer objetivo a minimizar se calcula como:

$$\text{mín } f_{Power}(s) = \sum_{i=1}^{|\mathcal{B}|} s_i \cdot P_i \tag{9}$$

donde P_i es el consumo de energía del SBS i (Eq. 8). Nótese que P_i incluye tanto la potencia de transmisión de cada celda contenida en i como la potencia de mantenimiento del SBS.

Sea $\mathcal{U}$ el conjunto de los UEs también desplegados, como se describe en la sección anterior, y $\mathcal{U}$ el conjunto de celdas en $\mathcal{B}$. Para calcular la capacidad total del sistema, los UEs se asignan primero a la celda activa que le proporciona la mayor SINR. Sea $\mathcal{A}(s) \in \{0,1\}^{|\mathcal{U}|\times|\mathcal{C}|}$ la matriz donde $a_{ij}^{s_j} = 1$ si $s_j = 1$ y la celda j sirve al UE i con la mayor SINR, y $a_{ij}^{s_j} = 0$ en caso contrario. Entonces, el segundo objetivo a maximizar, que es la capacidad total proporcionada a todos los UEs, se calcula como:

$$\text{máx } f_{Cap}(s) = \sum_{i=1}^{|\mathcal{U}|} \sum_{j=1}^{|\mathcal{C}|} a_{ij}^{s_j} \cdot BW_i^j \tag{10}$$

donde BW_i^j es el ancho de banda compartido de la celda j proporcionado al UE i (Ec. 7). Cabe destacar que estos dos objetivos del problema son claramente contradictorios entre sí, ya que la desconexión de las estaciones base conduce a una reducción del consumo de energía de la red, pero también perjudica la capacidad recibida por el usuario, debida al aumento de la distancia UE-Celda (incrementando las pérdidas de propagación), al mismo tiempo que se reduce el ancho de banda disponible para servir a los usuarios.

II-C. Multiconectividad

La MC es una técnica que surge como una generalización de la conectividad dual (Dual Connectivity, DC) en las redes 4G o LTE (Long Term Evolution). Consiste en la posibilidad de que un UE reciba simultáneamente señal de diferentes SBSs de distinta frecuencia. Actualmente, en organizaciones como 3GPP o ETSI, sólo contemplan la posibilidad de recibir señal de dos nodos diferentes, pero se prevee que este número aumente.

En esta conexión múltiple, se sigue una estructura maestro-esclavo donde uno de los nodos se comporta como nodo principal (Master node) y el resto como secundarios (Secundary nodes). Aunque existen diferentes planificaciones para la información que gestionan cada tipo de nodo, normalmente el nodo maestro se encarga tanto del plano de usuarios (datos)

como del plano de control, mientras que el papel de los nodos secundarios queda relevado al plano de usuarios. Por tanto, esta técnica permite mejorar la capacidad ofrecida a los usuarios o mejorar la fiabilidad en la conexión para aplicaciones URLLC (Ultra Reliability Low Latency Communications) [17].

En este trabajo, cada usuario puede recibir señal de dos o tres SBSs diferentes. Para ello, siguiendo la línea comentada anteriormente, primero cada usuario se conecta a la celda servidora que le ofrezca mayor SINR. A continuación, para asignar varias señales de distinta frecuencia provenientes de cada tipo de celda al UE, se aplica un umbral adaptativo según esta SINR máxima, $SINR_{MAX}$, ofrecida por la celda servidora y, calculado como:

$$SINR_j > SINR_{MAX} \cdot Th_{MC} \tag{11}$$

donde, $SINR_j$ es la SINR ofrecida por la celda j y Th_{MC}, el porcentaje de la $SINR_{MAX}$ que debe superar $SINR_j$. El valor de Th_{MC} se ha fijado 0.5. La finalidad de este umbral es la de evitar sumar señales que ofrezcan pocos beneficios a la capacidad del usuario y reducir la sobrecarga en la gestión de la red. Una vez asignadas las SBSs servidoras, al usar distinta frecuencia, no existen interferencias entre las distintas señales y, por tanto, se puede aproximar la capacidad total ofrecida al usuario como la suma de las capacidades ofrecidas por cada SBSs al UE. Adaptando la Ec. 6 para la recepción de varias señales, se obtiene:

$$C_k^j[bps] = \sum_{n=1}^{|\mathcal{N}|}(BW_{kn}^{j_n}[Hz] \cdot \sum_{i=1}^{r} \log_2\left(1 + \frac{SINR_{kn} \cdot \lambda_{in}}{n_{tx_n}}\right)) \tag{12}$$

donde N es el número total de señales que se asignan al UE.

III. Metaheurísticas multiobjetivo utilizadas: representación y operadores

Con el objetivo de evaluar el efecto de la multiconectividad en el espacio de búsqueda del apagado selectivo de SBS en UDNs, se han utilizado cuatro algoritmos bien conocidos de la literatura, que cubren tres de las aproximaciones más usuales para aproximar frentes de Pareto: técnicas basadas en dominancia (NSGA-II [10] y MOCell [11]), técnicas basadas en descomposición de objetivos (MOEA/D [12]) y técnicas guiadas por indicadores de calidad (SMS-EMOA [13]).

Las soluciones candidatas usan una representación clásica con cadena binaria de longitud L, siendo L el número de SBSs de la UDN, en el que cada bit representa el encendido o apagado ('1' o '0', respectivamente) de cada una de ellas. Dadas las densidades de despliegue mostradas en la Tabla I y el área de trabajo considerada, la longitud promedio de las soluciones para las instancias L, M y H son, respectivamente, 389.1, 682.5 y 993.32.

En cuanto a la configuración de los algoritmos, el tamaño de la población es de 100 individuos, y los operadores de selección, cruce y mutación son, respectivamente, *torneo binario*, *cruce de dos puntos* (DPX) con $c_r = 0.9$ y *bit flip* con $m_r = 1/L$. La condición de parada queda definida por el cómputo de 100000 evaluaciones de la función objetivo. Con el objetivo de evaluar de forma justa los motores de búsqueda, los cuatro algoritmos utilizados en este trabajo comparten la representación y los operadores, que están integrados en el framework para optimización multiobjetivo *jMetal*, donde también se ha implementado todo el modelado descrito en la sección anterior, y que está disponible para su descarga en `github.com/galeanobra/udn_multiconectivity`.

IV. Experimentación

En esta sección se describe la metodología utilizada en los experimentos, así como el análisis de los resultados obtenidos.

IV-A. Metodología

Para cada algoritmo, y buscando una comparación lo más justa posible, se han realizado 30 ejecuciones independientes, garantizando que las instancias de los diferentes escenarios de UDN son las mismas para todos, puesto que se generan aleatoriamente con el mismo conjunto de 30 semillas. La calidad de las aproximaciones al frente de Pareto se ha medido en base a dos indicadores de calidad. Por un lado, el Hypervolumen (HV) [18], un indicador *Pareto-compliant*, que mide el volumen cubierto por cada aproximación al frente de cada algoritmo. Dado que su valor es muy dependiente del escalado arbitrario de los objetivos, antes de calcularlo se ha procedido con un proceso de normalización con respecto a un frente de referencia (RPF) compuesto por todas las soluciones no dominadas encontradas por todos los algoritmos para un mismo escenario. El análisis estadístico de estos resultados no se incluye por falta de espacio.

Por otra parte, también se han utilizado las superficies de consecución (del inglés *attainment surfaces*) [19], que son una representación gráfica del rendimiento esperado en el espacio de objetivos. En este trabajo se ha usado la *50 %-attainment surface* que, informalmente, es la aproximación promedio al frente de Pareto, lo que sería equivalente a la media en el caso monoobjetivo. La fortaleza de esta herramienta es que aporta información sobre la distribución promedio de soluciones no dominadas en las distintas aproximaciones al frente de Pareto.

IV-B. Resultados

Comencemos primero mostrando los resultados obtenidos por el HV cuando no hay MC, y cuando hay MC con 2 y 3 portadoras, que se incluyen, respectivamente, en las Tablas II a IV. Los datos se corresponden a la mediana y rango intercuartílico (IQR) de las 30 ejecuciones, destacando con un fondo gris en cada fila el mejor algoritmo para cada escenario. Además del fondo gris, en el caso de MOEA/D también aparecen celdas de la tabla con un guión ($-$), y que indican que todas las aproximaciones al frente de Pareto de ese algoritmo para esa instancia están fuera del volumen contenido por el frente de referencia, es decir, su $HV = 0$.

La primera conclusión que se puede extraer de las tres tablas es que, de las 36 combinaciones (4 algoritmos $\times$ 9 instancias), el uso de la MC (ya sea con dos o tres portadoras) ha permitido alcanzar aproximaciones al frente de Pareto con un valor de

Tabla II
HV PARA UNA PORTADORA (MEDIANA$_{IQR}$)

	MOCell	MOEA/D	NSGA-II	SMS-EMOA
LL	$0.4077_{0.2382}$	$0.0373_{0.0895}$	$0.4171_{0.2400}$	$0.4102_{0.2310}$
LM	$0.4547_{0.2186}$	$0.1153_{0.1294}$	$0.4509_{0.1666}$	$0.4453_{0.2361}$
LH	$0.4707_{0.1992}$	–	$0.4839_{0.2452}$	$0.4691_{0.2207}$
ML	$0.4384_{0.2193}$	$0.0506_{0.1206}$	$0.4333_{0.2331}$	$0.4330_{0.2605}$
MM	$0.4948_{0.2609}$	$0.0606_{0.1395}$	$0.5146_{0.2657}$	$0.5157_{0.2692}$
MH	$0.5466_{0.2422}$	$0.0805_{0.1134}$	$0.5282_{0.2954}$	$0.5034_{0.2215}$
HL	$0.4790_{0.1939}$	–	$0.4763_{0.2198}$	$0.4553_{0.1959}$
HM	$0.5015_{0.2373}$	–	$0.5015_{0.2130}$	$0.4937_{0.2157}$
HH	$0.5129_{0.2306}$	–	$0.5070_{0.2239}$	$0.5022_{0.2381}$

Tabla III
HV PARA DOS PORTADORAS (MEDIANA$_{IQR}$)

	MOCell	MOEA/D	NSGA-II	SMS-EMOA
LL	$0.4234_{0.2202}$	$0.0283_{0.0637}$	$0.4196_{0.2179}$	$0.4222_{0.2444}$
LM	$0.4478_{0.1827}$	$0.0833_{0.1236}$	$0.4504_{0.1942}$	$0.4561_{0.2482}$
LH	$0.4608_{0.1859}$	–	$0.4832_{0.2086}$	$0.4758_{0.2098}$
ML	$0.4308_{0.2280}$	$0.0426_{0.0819}$	$0.4432_{0.2427}$	$0.4366_{0.2563}$
MM	$0.5085_{0.2681}$	$0.0442_{0.0923}$	$0.5208_{0.2592}$	$0.5080_{0.2432}$
MH	$0.5184_{0.2592}$	$0.0922_{0.1347}$	$0.5445_{0.2615}$	$0.5263_{0.2738}$
HL	$0.4732_{0.1965}$	–	$0.4755_{0.2046}$	$0.4641_{0.2173}$
HM	$0.5074_{0.2652}$	$0.0023_{0.0279}$	$0.5181_{0.2368}$	$0.4947_{0.2483}$
HH	$0.5161_{0.2313}$	–	$0.5130_{0.2046}$	$0.4997_{0.1972}$

Tabla IV
HV PARA TRES PORTADORAS (MEDIANA$_{IQR}$)

	MOCell	MOEA/D	NSGA-II	SMS-EMOA
LL	$0.4276_{0.2081}$	$0.0249_{0.0780}$	$0.4281_{0.2543}$	$0.4065_{0.2177}$
LM	$0.4497_{0.1661}$	$0.1287_{0.0825}$	$0.4261_{0.2088}$	$0.4360_{0.2330}$
LH	$0.4882_{0.2126}$	$0.0061_{0.0850}$	$0.4972_{0.2389}$	$0.4836_{0.1621}$
ML	$0.4370_{0.2529}$	$0.0385_{0.1060}$	$0.4302_{0.2654}$	$0.4473_{0.2508}$
MM	$0.5023_{0.2822}$	$0.0543_{0.0944}$	$0.5199_{0.3037}$	$0.5122_{0.2406}$
MH	$0.5279_{0.2442}$	$0.0754_{0.1041}$	$0.5274_{0.2658}$	$0.5201_{0.2303}$
HL	$0.4564_{0.2102}$	$0.0139_{0.0810}$	$0.4723_{0.1884}$	$0.4742_{0.1750}$
HM	$0.4892_{0.1794}$	$0.0357_{0.0840}$	$0.5116_{0.2245}$	$0.5138_{0.2060}$
HH	$0.4999_{0.2486}$	–	$0.5199_{0.2264}$	$0.5147_{0.2510}$

HV mayor (i.e., mejor) en 30. Las excepciones son MH y HL en MOCell, LL en MOEA/D, y LH y HL en NSGA-II. Desde el punto de vista del ingeniero de la red, la MC es, por tanto, un mecanismo que le va a permitir disponer de un conjunto de soluciones no dominadas entre las que elegir de mayor calidad. Hay 18 casos (50 %) en los que el uso de dos portadoras mejora a la configuración base (sin MC), otros 18 casos (diferentes de los anteriores) en los que esto ocurre con tres portadoras y, finalmente, 14 donde cualquier configuración con MC (dos o tres portadoras) obtiene un valor de HV más alto que el de la configuración base.

Vamos ahora al análisis detallado de cada tabla. En la configuración base, donde se aplica MC (Tabla II), se puede observar que MOCell es el algoritmo que destaca puesto que alcanza el mayor valor de HV para 5 de las 9 instancias, seguido de NSGA-II (3) y SMS-EMOA (1). MOEA/D ha obtenido, con diferencia, los peores valores (más bajos) del indicador, además, de forma consistente en todo el estudio, lo que nos lleva a concluir que la descomposición de objetivos, estrategia que usa para aproximar el frente de Pareto, no se ha integrado bien con la representación binaria y los operadores de mutación y cruce utilizados (bit flip y DPX) para este problema. Aún teniendo un modelado más preciso, los resultados de NSGA-II y MOCell son consistentes con los obtenidos en trabajos previos [6].

La introducción de la MC con una portadora extra (Tabla III) da, sin embargo, un vuelco a la comparativa, en el que ahora NSGA-II obtiene el mejor (mayor) HV en 6 escenarios, y MOCell en 2. Teniendo en cuenta que los despliegues de las UDNs son los mismos para cualquier número de portadoras (UEs y SBSs tienen idénticas posiciones en el área de trabajo), se puede observar que la MC tiene un efecto importante en el espacio de búsqueda y, por tanto, su exploración por parte de los algoritmos. Y es que la MC añade un grado adicional de epístasis, ya que apagar una celda ya no sólo tiene efecto en aquellos UEs para los que esa celda es su servidora, si no también para todos aquellos que la tienen como segunda celda alternativa. La portadora es la misma para todas las antenas de ese tipo en toda la red. En definitiva, el problema se vuelve más complejo. Uno de los casos extremos es, por ejemplo, MOCell para el escenario MH, cuyas aproximaciones promedio se muestran en la Fig. 1, y donde se puede observar que no usar MC proporciona soluciones de más capacidad y, en muchos casos, de menor consumo que cualquiera de las configuraciones en la MC activada. Como se ha comentado anteriormente, el uso de MC ayuda, en general, a encontrar mejores soluciones en base el HV, ya que proporcionan a los UEs una mayor capacidad (las aproximaciones dominan arriba y a la derecha en las gráficas), es decir, que, a igualdad de consumo la MC nos puede proporcionar soluciones con una mayor calidad de servicio. Este es precisamente el objetivo de esta tecnología.

Cuando se habilita el uso de una tercera portadora (Tabla IV) se observa de nuevo cambios en cuanto al comportamiento de los algoritmos, ya que no hay uno que destaque claramente. De hecho, sólo en 19 de las 38 combinaciones, el uso de tres portadoras mejora a las configuraciones con dos. Particularizando, hay casos como MOCell, en HM y HH, NSGA-II, en ML y MH, o SMS-EMOA en LL y LM, en el nivel de complejidad en el espacio de búsqueda que

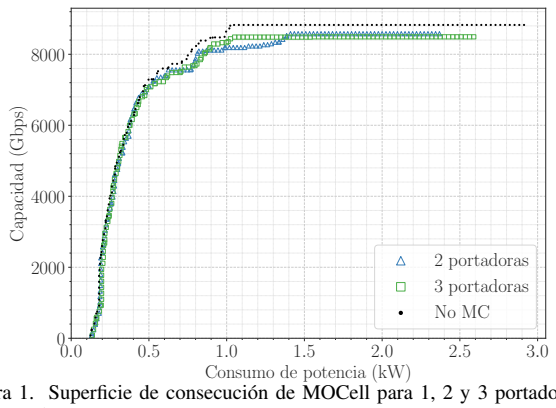


Figura 1. Superficie de consecución de MOCell para 1, 2 y 3 portadoras en el escenario MH

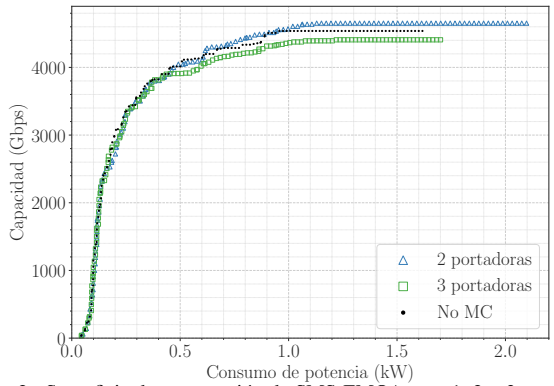


Figura 2. Superficie de consecución de SMS-EMOA para 1, 2 y 3 portadoras en el escenario LM

introduce una tercera portadora hace que el algoritmo empeore incluso a la configuración base sin MC. La Fig. 2 lo ilustra con la superficie de consecución de SMS-EMOA para LM. Con estas gráficas también queremos poner de manifiesto que, aunque los valores de HV están muy ajustados entre sí, realmente se deben al proceso de normalización que se usa para evitar que éste proporcione valores engañosos, pero que las pequeñas diferencias tienen un reflejo muy relevante en las aproximaciones al frente de Pareto obtenidas.

V. Conclusiones y trabajo futuro

Las redes ultra-densas son unos de los pilares fundamentales de los sistemas de quinta generación de redes móviles, pero incurren en un elevado consumo energético al tener en funcionamiento un elevando número de estaciones base, aún en periodos de baja demanda. Este problema se ha abordado previamente en la literatura como un problema de optimización multiobjetivo que consiste en el apagado selectivo de estas estaciones base, lo que reduce por un lado el gasto de energía, pero a la vez se ha de maximizar la capacidad del sistema a los usuarios de la red. Este trabajo analiza el impacto de usar la multiconectividad, una tecnología que permite a los usuarios recibir información de más de una estación base, en las soluciones no dominadas que alcanzan cuatro metaheurísticas multiobjetivo bien conocidas del dominio: NSGA-II, MOCell, MOEA/D y SMS-EMOA, que cubren diferentes metodologías para aproximar frentes de Pareto (dominancia, descomposición y guiadas por indicadores). Se ha diseñado una experimentación en la que, para el mismo conjunto de instancias, que representan nueve escenarios con distintos niveles de densificación, y la misma configuración de los algoritmos, se han abordado problemas sin multiconectividad, y con multiconectividad usando dos y tres portadoras. Los resultados obtenidos nos permiten concluir que la multiconectividad incrementa la complejidad de los espacios de búsqueda asociados al problema, puesto que introduce un mayor nivel de epístasis, i.e., encender/apagar una estación base tiene influencia en la capacidad que proporcionan el resto de estaciones base a los usuarios. Tanto es así que, para una misma instancia en la que se usa la multiconectividad, los algoritmos encuentran peores aproximaciones al frente de Pareto. Por otra parte, esta tecnología también permite mejorar en muchos escenarios las prestaciones de las redes ultradensas en cuanto a capacidad, pero que tiene escasa repercusión en la reducción del consumo energético de las mismas. Como trabajo futuro se plantea la hibridación de los algoritmos mediante operadores específicos que ayuden a mejorar la búsqueda de las distintas metaheurísticas multiobjetivo cuando se usa la multiconectividad, incorporando información del problema relativa a esta tecnología.

Referencias

[1] H. Tataria, M. Shafi, A. Molisch, M. Dohler, H. Sjoland, and F. Tufvesson, "6g wireless systems: Vision, requirements, challenges, insights, and opportunities," *Proceedings of the IEEE*, 2021.

[2] D. Lopez-Perez, M. Ding, H. Claussen, and A. H. Jafari, "Towards 1 Gbps/UE in Cellular Systems: Understanding Ultra-Dense Small Cell Deployments," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2078–2101, 2015.

[3] X. Ge, S. Tu, G. Mao, C. X. Wang, and T. Han, "5G Ultra-Dense Cellular Networks," *IEEE Wireless Comm.*, vol. 23, no. 1, pp. 72–79, 2016.

[4] 3GPP, "Small Cell Enhancements for E-UTRA and E-UTRAN—Physical Layer Aspects," tech. rep., 3GPP, 2014.

[5] D. Gonzalez G., J. Hamalainen, H. Yanikomeroglu, M. Garcia-Lozano, and G. Senarath, "A Novel Multiobjective Cell Switch-Off Framework for Cellular Networks," *IEEE Access*, vol. 4, pp. 7883–7898, 2016.

[6] F. Luna, R. Luque, J. Martínez, J. Valenzuela, and P. Padilla, "Addressing the 5G cell switch-off problem with a multi-objective cellular genetic algorithm," in *IEEE 5G World Forum*, pp. 422–426, 2018.

[7] P. Zapata-Cano, F. Luna, J. Valenzuela-Valdés, A. M. Mora, and P. Padilla, "Metaheurísticas híbridas para el problema del apagado de celdas en redes 5G (in Spanish)," in *XIII Congreso Español en Metaheurísticas y Algoritmos Evolutivos y Bioinspirados*, pp. 665 – 670, 2018.

[8] O. Alamu, A. Gbenga-Ilori, M. Adelabu, A. Imoize, and O. Ladipo, "Energy efficiency techniques in ultra-dense wireless heterogeneous networks: An overview and outlook," *Engineering Science and Technology, an International Journal*, 2020.

[9] K. Zhang, X. Xu, J. Zhang, B. Zhang, X. Tao, and Y. Zhang, "Dynamic multiconnectivity based joint scheduling of embb and urllc in 5g networks," *IEEE Systems Journal*, vol. 15, no. 1, pp. 1333–1343, 2021.

[10] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II," *IEEE transactions on evolutionary computation*, vol. 6, no. 2, pp. 182–197, 2002.

[11] A. J. Nebro, J. J. Durillo, F. Luna, B. Dorronsoro, and E. Alba, "MOCell: A cellular genetic algorithm for multiobjective optimization," *Int. J. of Intelligent Systems*, vol. 24, no. 7, pp. 726–746, 2009.

[12] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Transactions on evolutionary computation*, vol. 11, no. 6, pp. 712–731, 2007.

[13] N. Beume, B. Naujoks, and M. Emmerich, "SMS-EMOA: Multiobjective selection based on dominated hypervolume," *European Journal of Operational Research*, vol. 181, no. 3, pp. 1653–1669, 2007.

[14] *Performance Limits of Multiple-Input Multiple-Output Wireless Communication Systems*, ch. 1, pp. 1–47. John Wiley & Sons, Ltd, 2005.

[15] N. Piovesan, A. Fernandez Gambin, M. Miozzo, M. Rossi, and P. Dini, "Energy sustainable paradigms and methods for future mobile networks: A survey," *Computer Communications*, vol. 119, pp. 101–117, 2018.

[16] J. Son, S. Kim, and B. Shim, "Energy efficient ultra-dense network using long short-term memory," in *2020 IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1–6, 2020.

[17] M.-T. Suer, C. Thein, H. Tchouankem, and L. Wolf, "Reliability and latency performance of multi-connectivity scheduling schemes in multi-user scenarios," in *2020 32nd International Teletraffic Congress (ITC 32)*, pp. 73–80, 2020.

[18] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach," *IEEE transactions on Evolutionary Computation*, vol. 3, no. 4, pp. 257–271, 1999.

[19] J. Knowles, "A summary-attainment-surface plotting method for visualizing the performance of stochastic multiobjective optimizers," in *ISDA'05*, pp. 552–557, IEEE, 2005.

Influence of the Alternative Objective Functions in the Optimization of the Cyclic Cutwidth Minimization Problem*

*Note: The full contents of this paper have been published in the volume Lecture Notes in Artificial Intelligence 12882 (LNAI 12882)

Sergio Cavero
Universidad Rey Juan Carlos
Madrid, Spain
sergio.cavero@urjc.es

Eduardo G. Pardo
Universidad Rey Juan Carlos
Madrid, Spain
eduardo.pardo@urjc.es

Abraham Duarte
Universidad Rey Juan Carlos
Madrid, Spain
abraham.duarte@urjc.es

***Abstract*—The quality of the solutions to a combinatorial optimization problem is usually measured using a mathematical function, named objective function. This function is also used to guide heuristic procedures through the solution space helping to detect promising search directions (i.e., it helps to compare the quality of different solutions). However, this task becomes very hard when many solutions are evaluated with the same value by the objective function. This fact commonly occurs in either max-min/min-max optimization problems. In those situations, a key strategy relies on the introduction of an alternative objective function. This function helps to determine which solution is more promising when the compared ones achieve the same value of the original objective function. In this paper we study the Cyclic Cutwidth Minimization Problem (CCMP), which is an example of a min-max optimization problem. Particularly, we analyze the influence in the search of using alternative objective functions within local search procedures. Also, we propose two alternative objective functions for the CCMP and compare its performance against a previously introduced one. Finally, we explored the combination of more than one alternative function.**

***Index Terms*—Cyclic cutwidth, Graph Layout Problem, Combinatorial optimization, Flat landscape**

DOI: https://doi.org/10.1007/978-3-030-85713-4_14

Bayesian Optimization for Permutations Spaces: A preliminary approach

Daniel Jimenez*, Josu Ceberio*, and Jose A. Lozano*†
*University of the Basque Country (UPV/EHU), Donostia-San Sebastian, Spain
†Basque Center for Applied Mathematics (BCAM), Bilbao, Spain

***Abstract*—In the last decades, Bayesian Optimization has demonstrated to be a powerful alternative of approximate optimization for either black-box or expensive-evaluation problems. However, its application on the combinatorial spaces is not straight forward, and despite some few works, still there is room for research.**

In this paper, we focus in the context of permutation-based optimization problems, and propose a Bayesian Optimization algorithm for dealing with it. Particularly, we employ the random keys function which permits to map real-coded vectors in $\mathbb{R}^n$ to permutations in $\mathbb{S}_n$. In addition, a novel kernel function based on the Kendall's-τ distance is also presented. Conducted experiments on benchmark instances of the flowshop scheduling problem reveal that the line followed in this research opens interesting lines for future investigations.

***Index Terms*—Bayesian Optimization, Gaussian Process, Kernel function, Random key, Permutation problems**

I. Introduction

Bayesian Optimization (BO) [1]–[3] is a global optimization method designed to deal with expensive-to-evaluate black-box objective functions. Since Jonas Močkus published his work in 1975 [4], [5], there has been considerable development, both in terms of theoretical achievements and also applications to real-world problems [6], [7]. These developments have been mainly motivated by its use in the field of machine learning [8], where, for instance, the tuning the hyperparameters of a neural network [3] implies training the entire network with the obvious computational cost.

The philosophy of this approach is to calculate the extreme point of a function with the minimum possible evaluations. To that end, BO use all the information available to take the best decision possible when selecting the next point to evaluate.

The key idea of BO is the use of a surrogate model of the function to optimize, which is updated with the information provided by each evaluated point. This surrogate model is used to choose the next point to evaluate by means of what is called an acquisition function. This acquisition function explicitly balances exploration versus exploitation in the choice of the next point.

In spite of the previous references, the application of Bayesian Optimization strategies is not direct when approaching combinatorial optimization problems. The main of the problem relies on the definition of the probabilistic models used. Usually, Gaussian Processes are used to model the information observed so far, nevertheless, they are naturally defined on the continuous $\mathbb{R}^n$ domain, and their application on a combinatorial space is not straight forward. Not only that but analogous models with similar properties to Gaussian Processes have not been extensively published, which makes unable the development of analogous alternatives for the combinatorial spaces.

Notwithstanding the difficulties, recently a number of research works have considered using BO in the context of combinatorial optimization. Kim et al. [9] propose using random mapping functions which embed the combinatorial space into a convex polytope in a continuous space, and then, use standard approaches for optimizing. In [10], the authors propose a new Gaussian Process that permits to model the combinatorial space that uses a combinatorial graph. Another work by Wu et al. [11], proposes a Variational Bayesian Optimization method that combines variational optimization and continuous relaxations to the optimization of the acquisition function for Bayesian optimization.

Inspired by the recent works of BO in combinatorial optimization [9], [11]–[13], in this paper, we are interested in extending the usage of BO to a specific subset of NP-Complete combinatorial optimization problems; we refer to the permutation-based problems. One critical aspect in these problems is that solutions of the problem are naturally codified as permutations of objects. The mutual exclusivity constraints associated with that permutation nature makes these type of problems very challenging for most of the exact, heuristic and meta-heuristic algorithms [14]. Some well known problems such as the Travelling Salesman Problem (TSP), the Permutation Flowshop Scheduling Problem (PFSP) and Quadratic Assignment Problem (QAP).

The application of the strategies proposed in [9]–[13] to permutation-based problems is not direct (motivated mainly by the restrictions of permutations). As a result, in this paper, we carry out a preliminary approach, by using the random keys strategy [15], as a bridge between the real-coded space $\mathbb{R}^n$ where the Gaussian Process is defined, and the permutations space $\mathbb{S}_n$ that is the search space of the problem to optimize.

In order to evaluate the performance of the algorithm, some experiments on benchmark instances of the PFSP have been conducted. Obtained results reveal that optimizing the PFSP

This work has been partially supported by the Research Groups 2019-2020 (IT1244- 19) and the Elkartek Program (Project Code KK-2020/00049) from the Basque Government, PID2019-106453GA-I00/AEI/10.13039/501100011033 research project from the Spanish Ministry of Science, and the European Research Council H2020 (the EMPHATIC project).

by means of Bayesian Optimization is possible. Not only that but, depicted plots show that, when appropriately adjusted, BO is able to learn from the samples and find better solutions.

The remainder of the manuscript is organized as follows. In the next section an extensive background on Bayesian Optimization is introduced. Then, in Section III, the contribution of the paper, the usage of the Random Keys function in the context of BO is presented. Afterwards, an experimental study that illustrates the possibilities of BO is conducted in Section IV. A discussion about the performance of the proposed algorithm, and the future lines for research in the context of the paper are proposed in Section V. Finally, the paper is concluded in Section VI.

II. Bayesian Optimization

In the field of machine learning, Bayesian Optimization (BO) has demonstrated to be a very powerful strategy used for solve optimization problems. Given a space of possible solutions $\mathbb{R}^n$, the objective is to find the maximum or minimum (find the minimum is equivalent to find the maximum of the negative function) of a function

$$\begin{aligned} f : \mathbb{R}^{\mathbf{n}} &\longrightarrow \mathbb{R} \\ \mathbf{x} &\longrightarrow f(\mathbf{x}) \end{aligned}$$

where $\mathbf{x} = (x_1, ..., x_n) \in \mathbb{R}^n$. However, there is no information of how the function $f(\mathbf{x})$ looks like, and it is highly unlikely that is a well defined mathematical function that we can derive to find its extreme points (we can think on $f(\mathbf{x})$ as executing a simulator with the parameters $\mathbf{x}$).

Hence, in order to find the extremes, the unique choice is to evaluate some points, and based on the quality measured with f, approximate it. A trivial approach consists of sampling $\mathbf{x_1}, ..., \mathbf{x_m}$ random solutions from $\mathbb{R}^n$ and calculate its shape. As the size of m increases, we will have a more accurate approximation of what the function $f(x)$ is. Nonetheless, it is easy to see that it is not an efficient method for selecting samples since (1) the size of the search space might be incredibly large and heterogeneous (specially in the combinatorial case), and (2) it does not take into account the available information of the previous selected points.

As opposed to the previous method the BO uses the available information to define a probability distribution that models the quality of the possible solutions. Given a set of previously evaluated $\mathbf{x_1}, \ldots, \mathbf{x_{k-1}}$, the objective is to find a new point $\mathbf{x_k}$, that maximizes the probability of being the best possible solution $\mathbf{x}^*$ with the previous collected information, i.e. to find the $\mathbf{x}$ that maximizes $P(f(\mathbf{x}) = f(\mathbf{x}^*))$. Knowing the previously computed values $f(\mathbf{x_1}), ..., f(\mathbf{x_{k-1}})$.

$$\arg\max_{\mathbf{x} \in \mathbb{R}^n} \; P(f(\mathbf{x}) = f(\mathbf{x}^*) | f(\mathbf{x_1}), \ldots, f(\mathbf{x_k}))$$

With the previous information added, the BO can perform a smarter search and reduce drastically the number of evaluations.

Before continuing, in the next section, we are going to define what Gaussian processes are, since BO commonly uses them.

A. Gaussian Process

Gaussian process [16] is a stochastic process where any finite distribution have a multivariate normal distribution, that is to say, for any $n \in \mathbb{N}$, the random vector $\mathbf{x} = (x_1, \ldots, x_n)$ follows a multivariate normal distribution with mean $\mu = E(\mathbf{x})$ and covariance matrix $\Sigma = cov(\mathbf{x}, \mathbf{x})$, which is denoted by

$$\mathbf{x} \sim \mathcal{N}(\mu, \Sigma)$$

It can be seen as a generalization of the Gaussian probability distribution which defines a distribution over functions, and inference is taking place directly in the space of functions. Similar to multivaraite normal distribution, Gaussian process is completely specified by its mean and kernel functions (see Section II-B) defined by

$$\begin{aligned} m(\mathbf{x}) &= \mathbb{E}[f(\mathbf{x})], \\ k(\mathbf{x}, \mathbf{x}') &= \mathbb{E}[f(\mathbf{x}) - m(\mathbf{x}))(f(\mathbf{x}' - m(\mathbf{x}')], \end{aligned}$$

and is written as

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}'))$$

where for any finite distribution, it is expressed as a multivariate normal distribution

$$f(\mathbf{x}) \sim \mathcal{N}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}'))$$

B. Kernel function

In order to choose new sample points in a smart way, a measure of similarity between inputs and outputs is needed, for that, the kernel function is used. Given the domain of the solutions, let's say $\mathbb{R}^n$, the kernel function is defined as a function $k : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ where the result is a measure of how correlated the inputs are.

Kernel also is called generalized dot product, intuitively, is a way of computing the dot product of two vectors x and y in some feature space. Let us suppose we have a mapping $g : \mathbb{R}^n \to \mathbb{R}^m$ that brings our vectors in $\mathbb{R}^n$ to some feature space $\mathbb{R}^m$. Then the dot product of $\mathbf{x}$ and $\mathbf{y}$ in this space is $g(\mathbf{x})^T g(\mathbf{y})$. A kernel is a function k that corresponds to this dot product, in other words, $k(x, y) = g(\mathbf{x})^T g(\mathbf{y})$.

For example, let us consider a simple polynomial kernel $k(\mathbf{x}, \mathbf{y}) = (1 + \mathbf{x}^T \mathbf{y})^2$ with $\mathbf{x}, \mathbf{y} \in \mathbb{R}^2$. Assuming that $\mathbf{x} = (x_1, x_2)$ and $\mathbf{y} = (y_1, y_2)$, if we expand this expression: $k(\mathbf{x}, \mathbf{y}) = (1 + \mathbf{x}^T \mathbf{y})^2 = (1 + x_1 y_1 + x_2 y_2)^2 = 1 + x_1^2 y_1^2 + x_2^2 y_2^2 + 2 x_1 y_1 + 2 x_2 y_2 + 2 x_1 x_2 y_1 y_2$. It is interesting to note that this is a dot product between two vectors of higher dimension $(1, x_1^2, x_2^2, \sqrt{2} x_1, \sqrt{2} x_2, \sqrt{2} x_1 x_2)$ and $(1, y_1^2, y_2^2, \sqrt{2} y_1, \sqrt{2} y_2, \sqrt{2} y_1 y_2)$, while original vector were in $\mathbb{R}^2$, we perform a higher dimension dot product without explicitly visit it.

In what follows, we describe some classical and common kernel functions. The most simple one is the **Linear Kernel**,

it is given by the dot product of two vectors plus an optional constant c given by Eq. 1.

$$k(\mathbf{x_i}, \mathbf{x_j}) = \mathbf{x_i}^T \mathbf{x_j} + c \quad (1)$$

A bit more complex but widely used is the **Squared-Exponential Kernel** also knows as **Radial Basis Function** or **RBF** [16], probably the most common covariance function. It is mainly applied to model smooth functions. Its mathematical expression is given by Eq. 2

$$k(\mathbf{x_i}, \mathbf{x_j}) = \exp(-\frac{d(\mathbf{x_i}, \mathbf{x_j})^2}{2l^2}) \quad (2)$$

where l is a parameter to describe the function smoothness. If the Euclidean distance between x and y is big, the functions returns a value closed to 0. If the Euclidean distance between x and y is close, the function returns a value closed to 1.

Finally, another very well known Kernel function is **Matern kernel** [16], a generalization of the Squared Exponential kernel given by the following equation,

$$k(\mathbf{x_i}, \mathbf{x_j}) = \frac{1}{\Gamma(\nu)2^{\nu-1}} \left(\frac{\sqrt{2\nu}}{l} d(\mathbf{x_i}, \mathbf{x_j}) \right)^{\nu} K_\nu \left(\frac{\sqrt{2\nu}}{l} d(\mathbf{x_i}, \mathbf{x_j}) \right)$$

where ν is a parameter to control the smoothness, $d(\cdot, \cdot)$ denotes the Euclidean distance, $K_\nu(\cdot)$ stands for the modified Bessel function and $\Gamma(\cdot)$ is the gamma function.

C. Acquisition function

After obtaining the posterior distribution of the objective function given the previous t samples $D_{1:t} = \{\mathbf{x_n}, f(\mathbf{x_n})\}_{n=1}^t$, with the use of some acquisition function u, the BO select the most promising point (because the prediction is high, the uncertainty is high or both) to evaluate in the next iteration $t+1$. This is another optimization problem, since it is the problem of finding a maximum

$$\mathbf{x_{t+1}} = \arg\max_{\mathbf{x}} \; u(\mathbf{x}|D_{1:t})$$

Below we describe 3 of the most used acquisition functions. The first one is the **Improvement-Based acquisition function** which given a set of measures $D_{1:t}$ and $f(\mathbf{x}^+) = \arg\max f(\mathbf{x_i}), i = 1, \dots t$, the improvement-based acquisition function maximizes the probability of improvement over the actual maximum $f(\mathbf{x}^+)$, that is to say, the objective is to maximize the probability of $P(f(\mathbf{x}) \geq f(\mathbf{x}^+))$. To avoid continuous selection of infinitesimally close and slightly better points, a constant parameter c is added as a minimum improvement threshold.

$$\arg\max_{\mathbf{x}} \; P(f(\mathbf{x}) \geq f(\mathbf{x}^+) + c)$$

Another widely used acquisition function is the **Exploration-exploitation trade-off** which assumes that sometimes, it may be more efficient to explore regions with high uncertainty before continuing to exploit the proximity of $f(x^+)$, this algorithm makes a distinction by adding a term to enhance the exploration when relevant.

$$\arg\max_{\mathbf{x}} \; P(\mu(\mathbf{x}) - f(\mathbf{x}^+) - c) + \sigma(\mathbf{x}))$$

Finally, probably the most common acquisition function is the **Upper Confidence Bound** or **UCB** which balances exploration and exploitation through a single hyperparameter.

$$\arg\max_{\mathbf{x}} \; P(\mu(\mathbf{x}) + \kappa\sigma(\mathbf{x}))$$

High values of κ makes the acquisition function favour points with high variance, causing more exploration. Low values of κ will make the acquisition function favour points with high mean, causing more exploitation.

III. Bayesian Optimization on Permutation Spaces

The application of optimization techniques on permutations spaces is a difficult task, the candidate solutions have to fulfill 3 conditions that are hardly compatible with BO:

1) Given a solution $\mathbf{x} = (x_1, \dots, x_n)$, all its x_i elements must be integer.
2) Every $x_i,\ i = 1, \dots, n$ of the solution must be in the interval, $1 \leq x_i \leq n$.
3) No repetitions of elements are allowed for a solution $\mathbf{x}$. $\forall i, j \in \{1, \dots, n\},\ x_i = x_j \iff i = j$

To handle with it, in this paper, we propose the use of BO with Gaussian Process over a permutations represented by Random Keys [15]. The Random Key, is a way of representing a vector in $\mathbb{R}^n$ as a permutation $\mathbb{S}_n$. Given a vector in $\mathbb{R}^n$, its values are ranked to translate it into a $\mathbb{S}_n$ permutation, that is to say, the indexes of the ordered array form a permutation. For example, given a $\mathbf{v} = (0.23, 0.12, 0.76)$ in $\mathbb{R}^3$, the permutation associated with the Random Key is $\sigma = 213$. This allows the use of continuous space optimization techniques in the permutation spaces, while when evaluating the cost function is required, the translation is performed from $\mathbb{R}^n$ to $\mathbb{S}_n$. With this representation, we introduce our algorithm RK-BO, to solve permutation space problems by BO. The algorithm works as follows, given a set of previous observed t values $D_{1:t}$, the BO calculates the surrogate function and acquisition function naturally. The next point to evaluate D_{t+1} is selected by maximizing the acquisition function. However, when evaluating the point D_{t+1} is translated by the previously explained process from a $\mathbb{R}^n$ vector to a $\mathbb{S}_n$ permutation. This new point, along with its value, are saved and the execution of the BO continues until a stop criterion is reached.

IV. Experimental Study

In order to evaluate the performance of the proposed algorithm, in this section, we introduce an experimental study using the Permutation Flowshop Scheduling Problem (PFSP) as benchmark problem.

A. PFSP

The Permutation Flowshop Scheduling Problem (PFSP) is a classical optimization problem where, given a matrix of processing times $P = [p_{ij}]_{n \times m}$ of n jobs in m machines, the aim is to find the ordering of the jobs that minimizes the overall processing time of the jobs subject to some constraints. One job is made up of m operations where the jth operation of each job must be processed on machine j for a specific time

TABLE I
ARPD RESULTS FOR TAILLARD'S BENCHMARK INSTANCES. RESULTS IN BOLD DENOTE THE APPROACH WITH THE LOWEST ARPD

Instance	Best	RK-BO	Rand	Instance	Best	RK-BO	Rand	Instance	Best	RK-BO	Rand
$20 \times 5 - 0$	14033	**0.09**	0.15	$20 \times 10 - 0$	20911	**0.10**	0.14	$20 \times 20 - 0$	33623	**0.07**	0.10
$20 \times 5 - 1$	15151	**0.11**	0.11	$20 \times 10 - 1$	22440	**0.11**	0.12	$20 \times 20 - 1$	33623	**0.07**	0.10
$20 \times 5 - 2$	13301	**0.12**	0.15	$20 \times 10 - 2$	19833	**0.10**	0.15	$20 \times 20 - 2$	31587	**0.08**	0.10
$20 \times 5 - 3$	15447	**0.14**	0.15	$20 \times 10 - 3$	18710	**0.10**	0.14	$20 \times 20 - 3$	33920	**0.08**	0.10
$20 \times 5 - 4$	13529	**0.15**	0.17	$20 \times 10 - 4$	18641	**0.16**	**0.16**	$20 \times 20 - 4$	31661	**0.12**	**0.12**
$20 \times 5 - 5$	13123	**0.19**	0.20	$20 \times 10 - 5$	19245	**0.09**	0.13	$20 \times 20 - 5$	34557	**0.07**	0.09
$20 \times 5 - 6$	13548	**0.13**	0.16	$20 \times 10 - 6$	18363	0.18	**0.17**	$20 \times 20 - 6$	32922	**0.06**	0.10
$20 \times 5 - 7$	13948	**0.14**	0.16	$20 \times 10 - 7$	20241	**0.13**	0.14	$20 \times 20 - 7$	32412	**0.08**	0.10
$20 \times 5 - 8$	14295	**0.13**	0.18	$20 \times 10 - 8$	20330	**0.12**	0.14	$20 \times 20 - 8$	33600	**0.06**	0.10
$20 \times 5 - 9$	12943	**0.16**	0.18	$20 \times 10 - 9$	21320	**0.11**	0.13	$20 \times 20 - 9$	32262	**0.09**	0.10
$50 \times 5 - 0$	64803	**0.16**	0.25	$50 \times 10 - 0$	87207	**0.19**	0.21	$50 \times 20 - 0$	125831	**0.14**	0.18
$50 \times 5 - 1$	68062	**0.19**	0.25	$50 \times 10 - 1$	82820	**0.18**	0.23	$50 \times 20 - 1$	119259	**0.14**	0.20
$50 \times 5 - 2$	63162	**0.20**	0.25	$50 \times 10 - 2$	79987	**0.23**	0.27	$50 \times 20 - 2$	116459	**0.17**	0.22
$50 \times 5 - 3$	68226	**0.18**	0.23	$50 \times 10 - 3$	86581	**0.18**	0.22	$50 \times 20 - 3$	120712	**0.15**	0.18
$50 \times 5 - 4$	69392	**0.17**	0.22	$50 \times 10 - 4$	86450	**0.19**	0.22	$50 \times 20 - 4$	118184	**0.17**	0.21
$50 \times 5 - 5$	66841	**0.18**	0.26	$50 \times 10 - 5$	86637	**0.17**	0.21	$50 \times 20 - 5$	120703	**0.14**	0.19
$50 \times 5 - 6$	66258	**0.18**	0.24	$50 \times 10 - 6$	88866	**0.16**	0.20	$50 \times 20 - 6$	122962	**0.15**	0.18
$50 \times 5 - 7$	64359	**0.16**	0.27	$50 \times 10 - 7$	86824	**0.17**	0.21	$50 \times 20 - 7$	122489	**0.14**	0.18
$50 \times 5 - 8$	62981	**0.17**	0.24	$50 \times 10 - 8$	85526	**0.17**	0.21	$50 \times 20 - 8$	121872	**0.14**	0.18
$50 \times 5 - 9$	68898	**0.16**	0.21	$50 \times 10 - 9$	88077	**0.17**	0.22	$50 \times 20 - 9$	124064	**0.13**	0.18
$100 \times 5 - 0$	253713	**0.18**	0.23	$100 \times 10 - 0$	299431	**0.20**	0.23	$100 \times 20 - 0$	367267	**0.18**	0.21
$100 \times 5 - 1$	242777	**0.20**	0.25	$100 \times 10 - 1$	274593	**0.20**	0.27	$100 \times 20 - 1$	374032	**0.18**	0.20
$100 \times 5 - 2$	238180	**0.19**	0.25	$100 \times 10 - 2$	288630	**0.20**	0.24	$100 \times 20 - 2$	371417	**0.18**	0.20
$100 \times 5 - 3$	227889	**0.20**	0.25	$100 \times 10 - 3$	302105	**0.19**	0.23	$100 \times 20 - 3$	373822	**0.17**	0.20
$100 \times 5 - 4$	240589	**0.19**	0.25	$100 \times 10 - 4$	285340	**0.20**	0.25	$100 \times 20 - 4$	370459	**0.19**	0.21
$100 \times 5 - 5$	232936	**0.20**	0.26	$100 \times 10 - 5$	270817	**0.21**	0.26	$100 \times 20 - 5$	372768	**0.18**	0.21
$100 \times 5 - 6$	240669	**0.20**	0.24	$100 \times 10 - 6$	280649	**0.20**	0.26	$100 \times 20 - 6$	374483	**0.19**	0.21
$100 \times 5 - 7$	231428	**0.23**	0.27	$100 \times 10 - 7$	291665	**0.19**	0.23	$100 \times 20 - 7$	385456	**0.16**	0.19
$100 \times 5 - 8$	248481	**0.21**	0.25	$100 \times 10 - 8$	302624	**0.19**	0.22	$100 \times 20 - 8$	376063	**0.16**	0.19
$100 \times 5 - 9$	243360	**0.20**	0.27	$100 \times 10 - 9$	292230	**0.21**	0.25	$100 \times 20 - 9$	379899	**0.17**	0.20
$200 \times 10 - 0$	253713	**0.20**	0.22	$200 \times 20 - 0$	299431	**0.20**	0.21	$500 \times 20 - 0$	367267	**0.18**	0.19
$200 \times 10 - 1$	242777	**0.20**	0.23	$200 \times 20 - 1$	274593	**0.20**	0.22	$500 \times 20 - 1$	374032	**0.17**	0.19
$200 \times 10 - 2$	238180	**0.22**	0.24	$200 \times 20 - 2$	288630	**0.18**	0.20	$500 \times 20 - 2$	371417	**0.18**	0.19
$200 \times 10 - 3$	227889	**0.20**	0.23	$200 \times 20 - 3$	302105	**0.20**	0.22	$500 \times 20 - 3$	373822	**0.18**	0.19
$200 \times 10 - 4$	240589	**0.20**	0.23	$200 \times 20 - 4$	285340	**0.21**	0.23	$500 \times 20 - 4$	370459	**0.18**	0.19
$200 \times 10 - 5$	232936	**0.22**	0.24	$200 \times 20 - 5$	270817	**0.21**	0.23	$500 \times 20 - 5$	372768	**0.18**	0.20
$200 \times 10 - 6$	240669	**0.21**	0.24	$200 \times 20 - 6$	280649	**0.19**	0.22	$500 \times 20 - 6$	374483	**0.18**	0.19
$200 \times 10 - 7$	231428	**0.20**	0.24	$200 \times 20 - 7$	291665	**0.20**	0.21	$500 \times 20 - 7$	385456	**0.17**	0.19
$200 \times 10 - 8$	248481	**0.21**	0.24	$200 \times 20 - 8$	302624	**0.20**	0.22	$500 \times 20 - 8$	376063	**0.18**	0.19
$200 \times 10 - 9$	243360	**0.21**	0.24	$200 \times 20 - 9$	292230	**0.20**	0.21	$500 \times 20 - 9$	379899	**0.18**	0.20

without interruption. A job only can start on the jth machine when its $(j-1)$th operation has finished on the $(j-1)$ machine, and also the machine j is free.

Literature reports different objective functions, the objective is to find a sequence of jobs which minimizes the total flow time. The sequence of n jobs is codified as a permutation σ from $\{1, \cdots, n\}$ to $\{1, \cdots, n\}$, where $\sigma(i)$ is the rank assigned to job i and $\sigma\langle i\rangle$ denoted the job with rank i in σ with $i = 1, \cdots, n$.

$$F(\sigma) = \sum_{i=1}^{n} c_{\sigma\langle i\rangle, m} \tag{3}$$

where $c_{\sigma_{\langle i\rangle, j}}$ is calculated recursively by equation 4.

$$c_{\sigma_{\langle i\rangle, j}} = \begin{cases} p_{\sigma_{\langle i\rangle, j}} & i = j = 1 \\ p_{\sigma_{\langle i\rangle, j}} + c_{\sigma_{\langle i-1\rangle, j}} & i > 1, j = 1 \\ p_{\sigma_{\langle i\rangle, j}} + c_{\sigma_{\langle i\rangle, j-1}} & i = 1, j > 1 \\ p_{\sigma_{\langle i\rangle, j}} + \max\{c_{\sigma_{\langle i-\rangle, j}}, c_{\sigma_{\langle i\rangle, j-1}}\} & i > 1, j > 1 \end{cases} \tag{4}$$

B. *Experimental settings*

To perform the experimentation of this first approach of applying BO for solving permutation problems, we have selected the instances of the well known Taillard's benchmark [17]. Nine different configurations of jobs and machines have been considered, 20×5, 20×10, 20×20, 50×5, 50×10, 50×20, 100×5, 100×10, 100×20, 200×10, 200×20 and 500×20. The first number is the numbers of jobs to schedule since the second number is the amount of machines.

The performance of two algorithms was assessed: Random search and the proposed Bayesian Approach. The executions was repeated 10 times on each of the instances and we are left with only the best result.

With regard to the Bayesian algorithm, 200 iterations/samples were obtained. In addition, to set the values of the parameters l and κ of the kernel and acquisition function are calculated by testing which values makes the optimizer perform better. For the kernel l, different values were tried from 0.3 to 3.0 in steps of 0.3. For the acquisition function κ, values from 10^{-3} to 10^{2} going through all powers of 10.

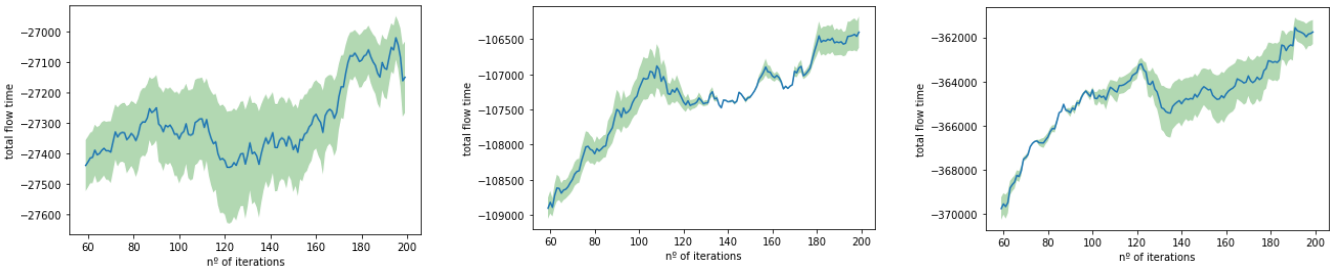


Fig. 1. Total flow time with respect to the number of iterations for the first Taillard instance of each of the sizes. At the left the results of size 20×10, at the middle 50×10 and at the right 100×20. The blue line and the green band represent the mean and standard deviation of the 10 runs. The y-axis negative values are because a maximization problem is performed, taking into account that, $min\ F(x)$ is equivalent to max -$F(x)$.

Based on the previous experiment, the selected values are $l = 1.2$ and $\kappa = 10.0$.

C. Results

The performance measure employed in our study is the average relative percentage deviation or ARPD

$$ARPD = \left(\sum_{i=1}^{10} \frac{(Algorithm_i - Best)}{Best} \right) /10$$

with $Algorithm_i$ being the solution found by the algorithm in the ith repetition and $Best$ the best known solution for the instance under study.

ARPD results of the experiments for instances of type 20×5, 20×10, 20×20, 50×5, 50×10, 50×20, 100×5, 100×10, 100×20, 200×10, 200×20 and 500×20 are introduced in Table I. Boldfaced numbers denote the best performing algorithm for each instance.

According to the results (see Table I), RK-BO obtained the best PFSP solution for 89 instances over 90. We can also observe in the Figure 1 how apparently as the number of iterations increases, the quality of the solutions increases too. This reinforces our hypothesis that with enough number of iterations, the global extreme will be found.

V. Discussion

The results obtained are promising, however, some problems have arisen. The use of a Random Keys representation introduces large redundancies, i.e., one permutation σ, has infinite Random Key representations. For example, $\sigma = 231$ can be represented as $\mathbf{x} = (0.3, 0.5, 0.1)$ or $\mathbf{y} = (0.4, 0.9, 0.2)$ among others. Furthermore, the euclidean distance between 2 vectors which represent the same permutation may not be 0, what would be a problem for BO. One possible solution for this is the use of a custom kernel, which uses a more appropriate distance function for permutations. For example, the Kendall distance could be used between the permutations associated to a $\mathbb{R}^n$ vectors.

We propose the use of a custom kernel given by Eq. 5, similar to the squared exponential kernel that we define by the Eq. 2, which uses the Kendall distance between permutations (represented as a Random Key) instead of the Euclidean distance.

$$k(x_i, x_j) = \exp\left(\frac{-d_k(R_K(x_i), R_K(x_j))}{2l^2} \right) \quad (5)$$

The Kendall distance d_k is a metric that counts the number of pairwise disagreements between two ranking lists.

$$\begin{aligned} d_k(\sigma_1, \sigma_2) = |\{(i,j) : i < j, \\ (\sigma_1(i) < \sigma_1(j) \wedge \sigma_2(i) > \sigma_2(j)) \vee \\ (\sigma_1(i) > \sigma_1(j) \wedge \sigma_2(i) < \sigma_2(j))\}| \end{aligned} \quad (6)$$

For example, given two permutations $\sigma_1 = (1, 3, 2)$ and $\sigma_2 = (3, 2, 1)$, the Kendall distance between them is the number of pairwise disagreements, that is to say, $(1, 3) \wedge (3, 2)$ and, $(1, 2) \wedge (3, 1)$, therefore, $d_k(\sigma_1, \sigma_2) = 2$.

VI. Conclusions

In this paper, we presented a preliminary work about the application of BO on permutation-based problems. The proposed approach consists of adapting the standard BO with Gaussian Processes by using the Random Keys functions to map real-coded vectors in $\mathbb{R}^n$ to $\mathbb{S}_n$. Such technique permits to convert any solution sampled with the acquisition function to an assessable permutation of the problem.

Conducted experiments on the PFSP showed that the algorithm is able to learn from the evaluated samples, and iteratively improve the quality of the obtained solutions. Results, however, suggest that the convergence of the proposed algorithm is slow, and the number of iterations needed to converge to good results is relatively high (which goes against the principle of application of BO).

In that sense, we think that there is a lot of room for future research, and discussed the implementation of kernel functions that take into account the permutation nature of the solutions based on metrics defined for that particular spaces.

It is noteworthy that the experiments carried out in the present work are preliminary, and more exhaustive work is deserved to better understand the dynamics of the studied algorithm when executing it on permutation problems. This is also a pending task for future research.

REFERENCES

[1] J. Mockus, *The Bayesian Approach to Local Optimization*. Dordrecht: Springer Netherlands, 1989, pp. 125–156. [Online]. Available: https://doi.org/10.1007/978-94-009-0909-0˙7

[2] E. Brochu, V. M. Cora, and N. de Freitas, "A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning," *CoRR*, vol. abs/1012.2599, 2010. [Online]. Available: http://arxiv.org/abs/1012.2599

[3] J. Snoek, H. Larochelle, and R. P. Adams, "Practical bayesian optimization of machine learning algorithms," 2012.

[4] J. Močkus, "On bayesian methods for seeking the extremum," in *Optimization techniques IFIP technical conference*. Springer, 1975, pp. 400–404.

[5] J. Mockus, "On the bayes methods for seeking the extremal point," *IFAC Proceedings Volumes*, vol. 8, no. 1, pp. 428–431, 1975.

[6] A. Baheri, S. Bin-Karim, A. Bafandeh, and C. Vermillion, "Real-time control using bayesian optimization: A case study in airborne wind energy systems," *Control Engineering Practice*, vol. 69, pp. 131–140, 2017.

[7] J. Song, Y. Chen, and Y. Yue, "A general framework for multi-fidelity bayesian optimization with gaussian processes," in *The 22nd International Conference on Artificial Intelligence and Statistics*. PMLR, 2019, pp. 3158–3167.

[8] A. Klein, S. Falkner, S. Bartels, P. Hennig, and F. Hutter, "Fast bayesian optimization of machine learning hyperparameters on large datasets," in *Artificial Intelligence and Statistics*. PMLR, 2017, pp. 528–536.

[9] J. Kim, M. Cho, and S. Choi, "Combinatorial bayesian optimization with random mapping functions to convex polytope," *arXiv preprint arXiv:2011.13094*, 2020.

[10] C. Oh, J. M. Tomczak, E. Gavves, and M. Welling, "Combinatorial bayesian optimization using the graph cartesian product," *arXiv preprint arXiv:1902.00448*, 2019.

[11] T. C. Wu, D. Flam-Shepherd, and A. Aspuru-Guzik, "Bayesian variational optimization for combinatorial spaces," *arXiv preprint arXiv:2011.02004*, 2020.

[12] G. Lan, J. M. Tomczak, D. M. Roijers, and A. Eiben, "Time efficiency in optimization with a bayesian-evolutionary algorithm," *arXiv preprint arXiv:2005.04166*, 2020.

[13] E. C. Garrido-Merchán and D. Hernández-Lobato, "Dealing with categorical and integer-valued variables in bayesian optimization with gaussian processes," *Neurocomputing*, vol. 380, pp. 20–35, 2020.

[14] J. Ceberio, "Solving permutation problems with estimation of distribution algorithms and extensions thereof," Ph.D. dissertation, Universidad del País Vasco-Euskal Herriko Unibertsitatea, 2014.

[15] M. Ayodele, J. McCall, O. Regnier-Coudert, and L. Bowie, "A random key based estimation of distribution algorithm for the permutation flowshop scheduling problem," in *2017 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, 2017, pp. 2364–2371.

[16] C. E. Rasmussen, "Gaussian processes in machine learning," in *Summer school on machine learning*. Springer, 2003, pp. 63–71.

[17] E. Taillard, "Benchmarks for basic scheduling problems," *European Journal of Operational Research*, vol. 64, no. 2, pp. 278 – 285, 1993, project Management anf Scheduling. [Online]. Available: http://www.sciencedirect.com/science/article/pii/037722179390182M

Analyzing Elementary Landscapes in Graph Partitioning Problems

Ignacio Dorado
UPV/EHU - University of the Basque Country
Donostia, Spain
nachodoradollamas@gmail.com

Josu Ceberio
University of the Basque Country (UPV/EHU)
Donostia, Spain
josu.ceberio@ehu.eus

***Abstract*—A fitness landscape of a combinatorial optimization problem is elementary when the objective function is an eigenfunction of the graph Laplacian induced by the neighborhood operator. This class of landscapes are characterized by a relative smoothness and decomposabilty. Among the list of problems that have been shown to generate a elementary landscape is the Min-Cut Graph Bipartition. However, different relaxations of the problem have not been addressed from this point of view yet.**

In this paper, we aim to expand the knowledge regarding the elementary nature of the Graph Partitioning Problem for its different variations by analyzing how it behaves when allowing multiple partitions and unbalanced sizes of them. We also investigate how elementary properties can be incorporated into algorithmic design in order to improve performance of local search based heuristics.

***Index Terms*—Graph Partitioning Problem, Elementary Landscapes, Local Search**

I. Introduction

Combinatorial optimization problems often belong to the NP-Hard family, so exact methods are usually computationally too expensive to be used in practice. Therefore, heuristic algorithms have been a recurrent alternative for such problems. Among these algorithms, the methods based on local search are a frequent choice. In this context, the theoretical analysis of the generated landscapes is specially relevant.

A fitness landscape for a combinatorial optimization problem is a triple (X, N, f). The search space X is the set of all feasible solutions for the problem and the objective function f is a map $f : X \to \mathbb{R}$ that associates every solution with a real objective value that measures how good that solution is. The goal is to find the element $x \in X$ that minimizes f. There is no loss of generality, since minimizing f is equivalent to maximizing $-f$. Finally, the neighborhood operator N is a function $N : X \to P(X)$, which assigns a set of neighbors $N(x)$ to a solution x. A solution y is a neighbor of x if $y \in N(x)$, we also say that y is reachable in one move from x.

Elementary landscapes were initially introduced when Grover [7] observed that some NP-Complete problems satisfy a simple equation similar to the wave equation in mathematical physics. This formula, known as Grover's wave equation, makes possible to compute the average objective value of the neighborhood of x without the need of evaluating every neighbor one by one:

$$\underset{y \in N(x)}{\text{avg}}\{f(y)\} = f(x) + \frac{c}{d}(\bar{f} - f(x)), \tag{1}$$

where d is the size of the neighborhood, $|N(x)|$, assuming a regular neighborhood; $\bar{f}$ is the average fitness value for all the solutions in the search space and c is a characteristic constant.

All landscapes satisfying this equation are considered elementary and share some unique properties. Grover [7] stated that every local maximum is greater than $\bar{f}$, every local minimum is smaller than $\bar{f}$, and the number of steps to reach a solution as good as $\bar{f}$ from any random initial solution is linear in problem size. This statement was supported by the results of Codenotti and Margara [6], which showed that the average value of $N(x)$ is always between $f(x)$ and $\bar{f}$, which also implies that certain types of plateaus cannot exists.

These properties result in a relative smoothness and decomposability of the landscape and allows local search methods to know in advance which neighborhoods are more promising than others. For this reason, many studies have been conducted on this topic, proving that certain number of problems belong to this class [7] [6] [11]. In addition, Chicano et al. [5] presented a methodology to decompose any objective function with a symmetric neighborhood into a superposition of elementary landscapes.

In this work, we are interested in the Graph Partitioning Problem (GPP), a well-known NP-hard problem with key applications in large variety of fields [2]. In his original publication, Grover [7] proved that, for the case of two partitions of the same size, and using a exchange of two vertices across the cut as neighborhood operator, the generated landscape satisfies Eq. 1. A similar proof was later carried out by Whitley and Sutton [12] [13] using a component based model. However, as far as we are aware, landscapes generated by multipartitions and variable size partitions have not been a focus of attention in the literature. In this paper, we aim to expand the knowledge on this matter, as well as some practical applications.

Following the component based analysis of the perfectly balanced bipartition problem [13], we provide a proof for the case of perfectly balanced multipartition and show that it also satisfies Grover's wave equation. We extend the investigation

to the case of unbalanced multipartition and show how particular instances of this variation do not always generate a elementary landscape.

In addition to the previous, in this paper we investigate different ways in which a local search based heuristic can be designed in order to benefit from elementary landscapes properties. Specifically, we incorporate the GPP specific knowledge into the design of local search algorithms.

The rest of the work is structured as follows. Sections II and III introduce the main concepts and terminology regarding elementary landscapes and Graph Partitioning problems. The incumbent study cases are presented in Sections IV, V and VI. Additionally, we introduced the proposed algorithms in Section VII and evaluate their performance in Section VIII. Finally, conclusion are summarized and new lines for future work are put forward in Section IX.

II. Elementary Landscapes

Grover's [7] wave equation was initially defined in terms of the difference operator ∇^2, which acts as a discrete equivalent of the continuous space Laplacian operator. Later, Stadler [10] developed a matrix version of the difference operator: the Laplacian matrix of the graph induced by the neighborhood operator, L_S. This matrix is calculated as $L_S = A - D$, where $A = [a_{xy}]_{|X|\times|X|}$ is the adjacency matrix with $a_{xy} = 1$ when $i \in N(j)$, and 0 otherwise. In addition, $D = [d_{xy}]_{|X|\times|X|}$ is the degree matrix with $d_{xy} = |N(x)|$ if $x = y$ and $d_{xy} = 0$ otherwise.

Since the objective function f can be characterized as a vector in $\mathbb{R}^{|X|}$, the Laplacian matrix of size $|X| \times |X|$ can be applied as a linear operator on f. This operation results in the neighborhood objective function value deviations:

$$(L_S f)(x) = \sum_{y \in N(x)} (f(y) - f(x)),$$

Stadler [10] showed that Eq. 1 holds for every neighborhood if, and only if, the objective function f, normalized at $\bar{f}$, is an eigenvector of L_S. Formally, if a constant eigenvalue (λ) exists such that $L_S(f - \bar{f}) = \lambda(f - \bar{f})$.

In the same line of research, Barnes et al. [1] pointed out that, although the definitions of Grover and Stadler were meant to be general and allow any type of neighborhoods, all their results assume only regular neighborhoods, i.e., $|N(x)| = d$ for a constant d and for all $x \in X$; and symmetric neighborhoods, i.e., $y \in N(x) \leftrightarrow x \in N(y)$ for all $x, y \in X$. Hence, they extended the notion of elementary landscape to irregular and non-symmetric neighborhoods using a new redefinition of the graph Laplacian: $L_B = I - D^{-1}A$. This makes it possible to characterize a more general class of problems satisfying Grover's wave equation; particularly, all the problems that satisfy $L_B(f - \alpha) = \lambda(f - \alpha)$ are elementary. It can be observed how the normalization factor α is not restricted to $\bar{f}$, but instead it can take any real value. Then, a more general wave equation can be considered:

$$\underset{y \in N(x)}{\text{avg}}\{f(y)\} = f(x) - \frac{\lambda}{d}(\alpha - f(x)), \qquad (2)$$

However, for any symmetric and regular neighborhood $\alpha = \bar{f}$. Consequently, it can be stated that Eq. 1 is a particular instance of Eq. 2 for symmetric and regular neighborhoods.

III. Graph Partitioning Problems

Given an undirected weighted graph $G = (V, E)$ with non-negative weights and a positive integer k, the Graph Partitioning Problem (GPP) [2] asks for a partition of G into k parts. More specifically, a solution is a collection of k disjoint subsets $V_0, ..., V_{k-1}$ that cover V, i.e., $V_i \cap V_j = \emptyset \ \forall i \neq j$ and $V_0 \cup ... \cup V_{k-1} = V$. A partition composed of 2 subsets or blocks of vertices is usually called bipartition or bisection, while one with k blocks is a k-partition. Furthermore, the cut is the number of edges between blocks or the total weight of those edges (capacity) if the graph is weighted. Although a number of different objective functions have been defined for the GPP, classically the aim in this problem is to find the partition of vertices in such a way that the cut size (or the weight of the edges in the cut) is minimized. For the rest of the work, this goal will be assumed.

A solution for the bipartition GPP (k=2) is naturally represented as a binary vector $\mathbf{x}$ of size n, where each position of this vector corresponds to the block that each vertex of the graph belongs to. This way, $x_i = 0$ if node v_i is in block 0 and $x_j = 1$ if node v_j is in block 1. Hence, the objective function to be minimized in the bipartition GPP can be expressed as

$$f(x) = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} w_{ij}(x_i(1 - x_j) + x_j(1 - x_i)), \qquad (3)$$

When the multipartition GPP is considered i.e., k partitions, then the previous objective function is generalized as follows:

$$f(x) = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} w_{ij}\mathbb{I}_{[x_i \neq x_j]}. \qquad (4)$$

The size of the blocks should also be monitored, either way the algorithm would most likely split the lowest degree node into a single group, which is not a good practical solution. For our purpose, the size of the blocks will come in as a constraint and will allow us to discern between two variations of the problem.

A. Perfectly Balanced GPP

This version of the GPP forces each block to have the exact same size, so it is restricted to instances where k is a factor of n. Nevertheless, this can be avoided by adding new nodes with no edges (edges of 0 cost for weighted graphs).

The case of $k = 2$ or Perfectly Balanced Bipartition has been a great focus of study in the literature and it has been shown to generate an elementary landscape under the exchange neighborhood operator [13] [7], which consist on swapping two nodes of different blocks across the cut.

B. Partially Balanced GPP

This more general relaxation of the problem is commonly referred as $(k, 1+\epsilon)$-balanced partitioning. It adds a balance constraint demanding that all subsets have about the same size. More specifically, each group contains at most $(1+\epsilon)\lceil n/k \rceil$ nodes. ϵ is usually referred as the imbalance parameter, the greater it is, the more imbalance is allowed in the partition. For $\epsilon = 0$, no imbalance is allowed, thus it behaves as a Perfectly Balanced GPP. As far as we have observed in the literature revision, there are no elementary landscape analysis on this problem.

Let us refer as Unbalanced GPP to any instance with no restriction on block sizes, apart from enforcing non-empty groups, which is the same as considering any $\epsilon \geq (n-k)(k-1)/n$.

IV. Case Study: Perfectly Balanced Bipartition

The Perfectly Balanced Bipartition Problem was first proven to be elementary under exchange neighborhood by Grover [7]. Later, Whitley and Sutton [13] introduced a new component based model and carried out a similar proof in a simpler manner, which is reproduced below.

Let us denote C to the set all the components that are used to compute the objective value of a solution. In this case, the set of components C is the set all weights, where $w_{i,j}$ is the weight associated to $e_{i,j} \in E$. Thus, $|C| = |E| = n(n-1)/2$. The objective function can always be expressed as a linear combination of the components.

If the components of C are uniformly distributed across the search space, then $\bar{f}$, is always computable. We know that each vertex v_i is connected to $n-1$ other vertices and $n/2$ out of those vertices are in a different block than v_i. This way we can compute $\bar{f}$ as

$$\bar{f} = \sum_{i=1}^{n-1}\sum_{j=i+1}^{n} w_{i,j}\frac{n/2}{n-1} = \frac{n}{2(n-1)}\sum_{i=1}^{n-1}\sum_{j=i+1}^{n} w_{i,j}. \quad (5)$$

The neighborhood operator will exchange a vertex in block 0 with a vertex in block 1. Thus, the size of the neighborhood d is the number of ways to pick one vertex from one group and then another one from the other group. Since each block has $n/2$ vertices, $d = n^2/4$. Knowing the objective value reduction produced by the leaving components in average due to the neighborhood operation, c_{out}, as well as the objective value average increase, c_{in}, it is possible to compute the average value for a full neighborhood in the following way:

$$\underset{y\in N(x)}{\text{avg}}\{f(y)\} = f(x) - \text{avg}\left\{\sum_{c\in c_{out}} c\right\} + \text{avg.}\left\{\sum_{c\in c_{in}} c\right\} \quad (6)$$

If $w_{i,j}$ is a weight contributing to the solution value ($e_{i,j}$ is in the cut), we can move v_i to the same block as v_j exchanging it with any of the other $n/2-1$ vertices. It is possible to do the same movements with v_j, so there is a number of $2(n/2-1) = n-2$ movements that remove the contribution of $w_{i,j}$ to the cut. Thus, the proportion of components leaving $f(x)$ after a neighborhood operation is $(n-2)/d$, as a result

$$\text{avg}\left\{\sum_{c\in c_{out}} c\right\} = \frac{n-2}{n^2/4}f(x) = \frac{4(n-2)}{n^2}f(x). \quad (7)$$

If $w_{i,j}$ is a weight not contributing to $f(x)$, it means v_i and v_j are in the same block. It is possible to exchange each one of them with any of the $n/2$ vertices in the other side. Then, there are $2(n/2)$ ways, out of d possible operations, in which $e_{i,j}$ can go inside the cut. Since the value of the components that are not in the cut is the total sum of all the components minus the value of those components in the solution $f(x)$, we have that

$$\begin{aligned}\text{avg}\left\{\sum_{c\in c_{in}} c\right\} &= \frac{n-2}{n(n-2)/4}\left(\sum_{i=1}^{n-1}\sum_{j=i+1}^{n} w_{i,j} - f(x)\right)\\ &= \frac{4}{n}\left(\frac{2(n-1)}{n}\bar{f} - f(x)\right), \end{aligned} \quad (8)$$

where the last step is a substitution from Eq. 5. Then, after some basic algebra we have that

$$\begin{aligned}\underset{y\in N(x)}{\text{avg}}\{f(y)\} &= f(x) - \frac{4(n-2)}{n^2}f(x) + \frac{4}{n}\left(\frac{2(n-1)}{n}\bar{f} - f(x)\right)\\ &= f(x) + \frac{2(n-1)}{n^2/4}(\bar{f} - f(x)) \\ &= f(x) + \frac{c}{d}(\bar{f} - f(x)), \end{aligned} \quad (9)$$

where $c = 2(n-1)$. Since Eq. 9 has the wave equation form for the Perfectly Balanced Bipartition GPP, then it is proved that the landscape generated under the exchange neighborhood is elementary.

V. Case Study: Perfectly Balanced Multipartition

In this section, we generalize the previous proof to the Perfectly Balanced Multipartition GPP. This time for a variable amount of blocks k. In addition, we show that it is possible to derive Grover's wave equation from components ratios.

In order to compute $\bar{f}$, it should be noticed that each vertex v_i is connected to $n-1$ other vertices. Since, there are $k-1$ groups of size n/k that do not contain v_i, each vertex is connected to $(n/k)(k-1)$ vertices from a different block. This way, it is possible to express the average fitness value as

$$\bar{f} = \sum_{i=1}^{n-1}\sum_{j=i+1}^{n} w_{i,j}\frac{(n/k)(k-1)}{n-1} = \frac{n(k-1)}{k(n-1)}\sum_{i=1}^{n-1}\sum_{j=i+1}^{n} w_{i,j}. \quad (10)$$

In this problem, the exchange neighborhood is a exchange between two vertices of two distinct blocks. So the number of possible exchanges d is given by the number of combinations of 2 blocks times the number of combinations of 2 vertices from those blocks. Since there are k blocks of size n/k:

$$d = \frac{k(k-1)}{2}\left(\frac{n}{k}\right)^2 = \frac{(k-1)n^2}{2k}.$$

Now, we calculate the average value of all the components that are leaving the solution after each neighborhood operation, c_{out}. If $w_{i,j}$ is a weight contributing to the objective value of the solution, then v_i and v_j are in different blocks. Exchanging v_i with any of the other $(n/k)-1$ vertices from the same block than v_j leads to a solution where $w_{i,j}$ is not in the cut anymore. By symmetry, the same reasoning applies to v_j. Thus, there are $2((n/k)-1)$ movements, out of d, removing $w_{i,j}$ contribution to the objective function value. Simplifying:

$$\text{avg}\left\{\sum_{c\in c_{out}} c\right\} = \frac{2(\left(\frac{n}{k}\right)-1)}{\frac{(k-1)n^2}{2k}} f(x) = \frac{4(n-k)}{n^2(k-1)} f(x). \quad (11)$$

If $w_{i,j}$ is a weight not contributing to $f(x)$, it means v_i and v_j are in the same block. To pull them apart, it is enough to exchange any of them with any one of the $n-(n/k)$ vertices in a different block. Consequently, $2(n-(n/k))$ movements out of d are yielding to a solution where $w_{i,j}$ is contributing to $f(x)$. Recall that the value of the components outside solution $f(x)$ is the value of all the components minus $f(x)$. Then:

$$\begin{aligned}\text{avg}\left\{\sum_{c\in c_{in}} c\right\} &= \frac{2\left(n-\frac{n}{k}\right)}{\frac{n^2(k-1)}{2k}} \left(\sum_{i=1}^{n-1}\sum_{j=i+1}^{n} w_{i,j} - f(x)\right) \\ &= \frac{4}{n}\left(\frac{k(n-1)}{n(k-1)}\bar{f} - f(x)\right), \end{aligned} \quad (12)$$

by substituting the sum of weights from Eq. 10.

Below it is shown how this development ends up in Grover's wave equation by Eq. 6.

$$\begin{aligned}\underset{y\in N(x)}{\text{avg}}\{f(y)\} &= f(x) - \frac{4(n-k)}{n^2(k-1)} f(x) + \frac{4}{n}\left(\frac{k(n-1)}{n(k-1)}\bar{f} - f(x)\right) \\ &= f(x) + \frac{2(n-1)}{\frac{n^2(k-1)}{2k}}(\bar{f} - f(x)) \\ &= f(x) + \frac{c}{d}(\bar{f} - f(x)), \end{aligned} \quad (13)$$

where $c = 2(n-1)$. As it can be seen, the equation is a generalization of the one previously shown for the Perfectly Balanced Bipartition and it is consistent with the existing results.

A. Partial Neighborhoods

It is not hard to see that the exchange neighborhood contains ineffective operations. Let us assume vertex v_i is in a different block than vertex v_j and neither of them has any edge going outside its block. Then, swapping v_i with v_j cannot reduce the objective value, but it can certainly increase it. Therefore, it would be desirable to avoid these non-improving movements and stick to a more promising partial neighborhood $N'(x)$, where $\underset{y\in N'(x)}{\text{avg}}\{f(y)\} <= \underset{y\in N(x)}{\text{avg}}\{f(y)\}$. Whitley and Sutton [13] showed that it is possible to efficiently compute the average value of particular partial neighborhoods using the component based model. In particular, they constructed a partial neighborhood component analysis for the Perfectly Balanced Bipartition. Following, we generalize the former proof for multipartitions problems.

According to Whitley and Sutton [13], the average objective value for the previously described partial neighborhood can be computed as:

$$\underset{y\in N'(x)}{\text{avg}}\{f(y)\} = f(x) - \frac{c}{d'} f(x) + \frac{\beta\left(\sum_{i=1}^{n-1}\sum_{j=i+1}^{n} w_{ij}\right) - W'}{d'}, \quad (14)$$

where d' is the size of $N'(x)$, W' is the sum of all weights that are no longer contribute to the partial neighborhood, and β is the number of different neighborhood operations in which an arbitrary component that does contribute to $f(x)$ can contribute to the objective function. Recall that $\beta = n$ for bipartition and $\beta = 2(n-n/k)$ for multipartition.

In order to compute d', we remove all non-improving movements from $N(x)$. Let n_i be the set of vertices in block i that do not have any edge connecting them with a different block. Then,

$$d' = d - \sum_{i=0}^{k-1}\sum_{j=i+1}^{k-1} |n_i||n_j|.$$

For every pair of blocks i, j, every move exchanging vertice x, y with $x \in n_i$ and $y \in n_j$ is not contemplated in the partial neighborhood. Thus, for each one of these moves, we remove all the edges incident on x and y from the total sum of weights. Therefore, W' is calculated as

$$W' = \sum_{i=0}^{k-1}\sum_{j=i+1}^{k-1}\left(|n_i|\sum_{v\in V, x\in n_j} w_{v,x} + |n_j|\sum_{v\in V, x\in n_i} w_{v,x}\right).$$

This way, it is possible to compute the average value of a partial neighborhood for a multipartition problem in a closed an efficient manner. Consequently, it is sometimes possible to know if a partial neighborhood contains an improving move.

VI. Case Study: Unbalanced Bipartition

Unlike the previous cases of study, the unbalanced scenario introduces some particularities that do not permit following the previous methodology for proving the elementariness of the landscapes. For Unbalanced GPP, the exchange neighborhood does not permit exploring solutions with different block sizes. In order to have a connected landscape of solutions, the neighborhood operator must enable moving between solutions with different block sizes. In that sense, we adopt a restricted version of the Hamming neighborhood operator, which moves one vertex to a different block at each step. The restriction is subject to avoiding movements that produce blocks with 0 vertices. Such restriction, however, leads to irregular neighborhoods, i.e. $|N(x)|$ is not constant for every $x \in X$. Most research in elementary landscapes focus only on regular neighborhoods. In fact, we are not aware of any elementary landscape generated under this type of neighborhood.

Lacking a proper methodology to find elementary landscapes with irregular neighborhoods, we restrict the study to analysing specific instances of the Unbalanced GPP. In this

line of research, Barnes et al. [1] argue that a landscape is elementary if, and only if,

$$L_B(f - \alpha) = \lambda(f - \alpha),$$

with $L_B = I - D^{-1}A$ and a constant α. Since the Laplacian is a stochastic matrix $L_B(f - \alpha) = L_B f$. Then, for any given GPP instance, it is possible to compute $L_B f$ and find its eigenvalues. This way, the generated landscape is elementary if, and only if, the system $L_B f = \lambda(f - \alpha)$ has a solution for any of the eigenvalues λ.

With the aim of gaining intuition about the elementariness of landscapes in this case of study, we approached the problem empirically, created artificial random graphs, and evaluated if the system $L_B f = \lambda(f - \alpha)$ has a solution. Specifically, we use the artificial benchmark by Johnson et al. [8] criterion to generate random graphs. This set of instances consists of random graphs $G_{n,p}$, where n is the number of vertices and an edge is randomly generated between each pair of vertices with probability p.

In the conducted experiment, 100 instances of random graphs of size 4 were generated, as well as 100 instances of size 6. Greater sizes were discarded due to time limitations, since the problem scales very quickly. The probability p was randomly chosen for each instance. None of the instances were proven to have a solution for the above-mentioned system, with two exceptions: empty instances with no edges and complete graphs of size 4. These results suggest that the landscape generated by Unbalanced GPP under restricted hamming distance is usually non-elementary.

VII. Proposed Algorithms

Although the study of elementary landscapes has been exhaustive through decades, algorithms that benefit from the problem specific knowledge are very limited [4]. In this sense, in what follows, we implement 4 different local search based algorithmic alternatives [1] for the Perfectly Balanced GPP (inspired in Lu et al. [9]).

Essentially, we can distinguish between to types of elementary properties: implicit and explicit [9]. Implicit properties, such as smoothness, can be beneficial for performance but can not be applied in the design of an algorithm. In the other hand, explicit properties can be taken into account when designing a strategy, such as computing the average of a neighborhood or the fact that every local maximum is greater than $\bar{f}$ and every local minimum is lower than $\bar{f}$.

Classical local search design usually ranges between two selection criteria of the solutions in the neighborhood: a Best First Search (BFS) chooses the first improving neighbor, while a Greedy Search (GS) look across the entire neighborhood and chooses the most improving neighbor. Those two algorithms are chosen as starting point for the following:

1) Elementary Best First Search (EBFS): every step chooses the first neighbor with a better average objective value of the solutions in its own neighborhood.
2) Elementary Greedy Search (EGS): every step looks through the entire neighborhood and chooses the neighbor with the best average objective value of the solutions in its own neighborhood.
3) Elementary Sampling Search (ESS): behaves as a BFS and takes advantage of the fact the average of $N(x)$ is always between $f(x)$ and $\bar{f}$. When $f(x) > \bar{f}$, the neighborhood is guaranteed to include an improving operation. Since the neighborhood is more promising, it is safe to sample n random solutions from $N(x)$ and take the one with best objective value. On the other hand, when $f(x) < \bar{f}$, taking a random set of samples is less promising so the first improving neighbor is chosen.
4) Elementary Partial Neighborhood Search (EPNS): when $f(x) > \bar{f}$ it behaves as ESS. Alternatively, when $f(x) < \bar{f}$, it expands a partial neighborhood $N'(x)$ and compute its average. If the average of $N'(x)$ is greater than $f(x)$, then it samples n solutions from $N'(x)$ and chooses the one with best objective value. Otherwise, it goes for the first improvement.

Note than EBFS and ESS are not guaranteed to converge at a local optima. If x^* is a local minima, then $\operatorname{avg}_{z \in N(y)}\{f(z)\} < \operatorname{avg}_{z \in N(x^*)}\{f(z)\}$ is true for every $y \in N(x^*)$, but not the other way around.

VIII. Computational Study

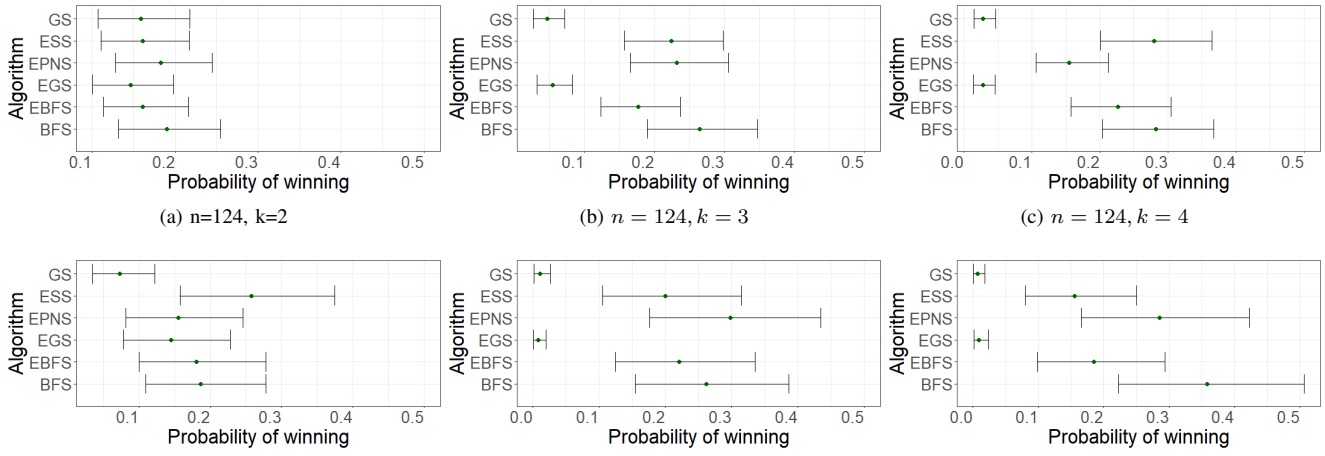
In order to assess the performance of the algorithms introduced in the previous section, 6 instances from the artificial benchmark of random graphs by Johnson et al. [8] are used. This set of instances consists of random graphs $G_{n,p}$, where n is the number of vertices and an edge is randomly generated between each pair of vertices with probability p. 10 repetitions were performed for each algorithm and instance, using as stopping criterion a maximum of $10n^2$ evaluations. We compare the results obtained for different number of partitions by the algorithms presented in Section VII, with the more general BFS and GS in Table I.

The table depicts a similar performance for all the algorithms when the number of partitions is low. However, increasing k seems to benefit BFS and ESS.

Additionally, we establish a performance ranking for the algorithms using Bayesian Performance Analysis [3]. In order to reduce uncertainty and better discern performance gaps, we artificially generated random graphs according to Johnson et al. [8] benchmark. Altogether, 28 random graphs with with 124 vertices and 12 random graphs with 250 vertices were evaluated. The probability of winning for each one of the algorithms is depicted in Fig. 1.

This results, in fact, remark the previous observations: BFS and ESS perform better for a higher number of partitions, while greedy algorithms fall a bit behind when dealing with a greater k and a greater n. In more complex problems, a greedy algorithm needs to evaluate more neighbors at each step, thus they reach the maximum number of evaluations being further from the local optimum.

[1] the code is openly available at https://github.com/nachodll/GPP-EL-Based-Heuristics.git



(a) n=124, k=2 (b) $n = 124, k = 3$ (c) $n = 124, k = 4$

(d) $n = 250, k = 2$ (e) $n = 250, k = 3$ (f) $n = 250, k = 4$

Fig. 1: Credible intervals of the statistical analysis on the 6 algorithms compared for $k = 2, 3, 4$ on instances with 124 vertices (p=0.01, 0.02, 0.04, 0.08, 0.16, 0.32, 0.64) and 250 vertices (p=0.01, 0.02, 0.04).

TABLE I: Medians of the best objective function value reached by the 6 algorihtms on the following instances.

Instance	BFS	GS	EBFS	EGS	ESS	EPNS
			k = 2			
G124.02	23.5	**22.5**	25.0	25.5	25.0	24.5
G124.16	476.5	479.5	477.0	473.0	474.5	**466.0**
G250.01	62.5	64.0	61.0	**59.0**	63.5	60.0
G250.02	**145.5**	149.0	148.5	148.5	146.5	**145.5**
G250.04	400.5	405.5	**390.0**	410.0	400.0	397.5
G250.08	**865.5**	891.0	870.0	891.5	870.0	881.0
			k = 3			
G124.02	36.0	35.5	36.5	**34.5**	36.0	33.5
G124.16	656.0	658.0	651.5	665.0	**649.5**	654.5
G250.01	**83.5**	108.5	86.0	108.0	85.5	85.0
G250.02	206.0	255.5	208.0	255.5	210.0	**205.0**
G250.04	548.5	627.5	**540.5**	634.0	550.5	549.5
G250.08	**1189.5**	1316.0	1200.0	1316.0	1210.0	1206.5
			k = 4			
G124.02	**42.0**	44.5	44.0	42.0	42.5	44.0
G124.16	751.0	774.5	752.5	771.5	**746.0**	750.0
G250.01	98.5	145.0	101.0	142.5	**97.0**	103.0
G250.02	**239.5**	315.5	241.5	315.5	240.5	240.5
G250.04	**634.0**	760.5	635.0	760.0	640.0	638.5
G250.08	1380.0	1551.5	1375.0	1549.5	**1371.5**	1374.0

IX. Conclusions and Future Work

In this paper, we constructed a component based analysis for the GPP, allowing a variable k, as well as a way to compute the average objective value of a partial neighborhood for any graph multipartition. We also explore different ways of designing problem specific algorithms with a explicit use of elementary properties and conducted a preliminary experimentation.

Yet, extensive experimentation is still needed in order to get a better understanding of elementary properties implication in algorithmic design. A formal study for the Partially Balanced GPP elementariness is also left for future research. We believe it might promising target for Elementary Landscape Decomposition.

Acknowledgment

This work has been partially supported by the Research Groups 2019-2020 (IT1244- 19) and the Elkartek Program (Project Code KK-2020/00049) from the Basque Government, PID2019-106453GA-I00/AEI/10.13039/501100011033 research project from the Spanish Ministry of Science.

References

[1] J. W. Barnes, B. Dimova, S. P. Dokov, and A. Solomon, "The theory of elementary landscapes" App. Math. Lett., 16(3):337–343, 2003.
[2] A. Buluç, H. Meyerhenke, I. Safro, P. Sanders, and C. Schulz, "Recent Advances in Graph Partitioning" Algor. Eng., 117–158, 2016.
[3] B. Calvo, O. M. Shir, J. Ceberio, C. Doerr, H. Wang, T. Bäck, and J. A. Lozano, "Bayesian performance analysis for black-box optimization benchmarking," Proc. GECCO'19, 2019.
[4] J. Ceberio, B. Calvo, A. Mendiburu, and J. Lozano, "Multi-Objectivising Combinatorial Optimisation Problems by Means of Elementary Landscape Decompositions" Evol. Comp., 27(2):291–311, 2019.
[5] F. Chicano, L. D. Whitley, and E. Alba, "A Methodology to Find the Elementary Landscape Decomposition of Combinatorial Optimization Problems," Evol. Comp., 19(4):597–637, 2011.
[6] B. Codenotti and L. Margara, "Local properties of some NP-complete problems," TR-92-021, Inter. Comput. Sci. Inst., Berkley, CA, 1992.
[7] L. K. Grover, "Local search and the local structure of NP-complete problems" Oper. Res. Lett., 12(4):235–243, 1992.
[8] D. S. Johnson, C. R. Aragon, L. A. McGeoch, and C. Schevon, "Optimization by Simulated Annealing: An Experimental Evaluation; Part I, Graph Partitioning," Oper. Res., 37(6):865–892, 1989.
[9] G. Lu, R. Bahsoon, and X. Yao, "Applying Elementary Landscape Analysis to Search-Based Software Engineering," 2nd Int. Symp. on Search Based Software Eng., 2010.
[10] P. F. Stadler, "Towards a theory of landscapes," Complex Systems and Binary Networks, 78–163.
[11] P. F. Stadler, "Landscapes and their correlation functions," J. Math. Chem., 20(1):1–45, 1996.
[12] L. D. Whitley, A. M. Sutton, and A. E. Howe, "Understanding elementary landscapes" Proc. 10th - GECCO '08, 2008.
[13] L. D. Whitley and A. M. Sutton, "Partial neighborhoods of elementary landscapes," Proc. 11th - GECCO '09, 2009.

A Matheuristic Approach to the Interval Job Shop Scheduling Problem

Sezin Afşar
Department of Computer Science
University of Oviedo
Gijón, Spain
afsarsezin@uniovi.es

Juan José Palacios
Department of Computer Science
University of Oviedo
Gijón, Spain
palaciosjuan@uniovi.es

Camino R. Vela
Department of Computer Science
University of Oviedo
Gijón, Spain
crvela@uniovi.es

***Abstract*—In this work, we tackle a variant of the job shop scheduling problem where uncertain processing times are modeled with intervals. The objective is to minimize the makespan by means of a ranking method that targets to minimize its midpoint and width. A matheuristic algorithm is proposed which is composed of a genetic algorithm and a constraint programming model. The experimental analysis shows the potential of the approach, obtaining makespan midpoint values that improve the state-of-the-art with narrower makespan intervals. A synergy analysis also shows the benefit of combining these two approaches.**

***Index Terms*—interval scheduling, matheuristic, constraint programming, genetic algorithm**

I. Introduction

Scheduling is a widely known optimization problem that arises in many industrial applications such as manufacturing and production as well as service applications such as information processing, transportation and distribution [22]. One of the most widely used variants of the problem is the *job shop scheduling* (JSP), since it is a good model for many real-life scenarios, while posing a challenge due to its complexity.

Although a large part of the literature is devoted to deterministic scheduling, uncertainty in design parameters such as processing times is common due to unexpected perturbations in real-life applications [21]. Uncertainty in task durations is one of the most common sources of uncertainty. For instance, depreciation in machinery over years of operation or the experience level variances among different workers leads to a range of possible durations. This type of uncertainty is often modeled using a stochastic approach which requires a prior knowledge of the probability distribution of the variables [22]. When there is lack of such information, uncertain task durations can be modeled using fuzzy numbers or fuzzy intervals which have possibility distributions representing more or less plausible values [7]. A third alternative for uncertain task durations is the interval model. In this approach, required prior knowledge is minimal, and therefore, it can be also seen as a first step towards modeling uncertainty.

Although modeling uncertainty with intervals seems a natural approach, literature on the job shop scheduling problem with interval durations (also referred as interval job shop problem, or IJSP in short) is limited. In [13], the authors propose a population-based neighborhood search (PNS) to optimize the makespan of a IJSP. More recently, a genetic algorithm is proposed in [5] to solve the same problem. In [14], the authors minimize total tardiness in an IJSP with interval due dates and propose a genetic algorithm to solve it. Another algorithm is then proposed in [6], improving the results from [14]. Multiobjective variants of the IJSP are also tackled in [15] and [17]. Interval durations are not present only in JSP. For instance, they are considered in a flexible job shop problem in [16], where a shuffled frog-leaping algorithm is utilized to minimize the makespan, and uncertain processing times are modeled as intervals in other scheduling problems in [1], [10], [19], [23].

Matheuristic algorithms are an increasingly popular approach in scheduling. They are obtained by inter-operating a metaheuristic algorithm with a mathematical model [3], [24]. The first provides the necessary diversity, while the latter provides high exploitation capabilities. This approach has been used among others, to tackle a bi-objective job shop problem that minimizes total weighted tardiness and energy consumption in [9], where a memetic algorithm that integrates a linear programming post-processing to decrease the energy consumption of a set of non-dominated solutions is used. Mathematical programming is also incorporated into metaheuristics to improve local search in [11] to solve relaxed versions of the problem in [18] or some sub-problems in [8].

We tackle the interval job shop scheduling problem with makespan minimization. To solve it, we propose a matheuristic approach by combining a state-of-the-art genetic algorithm with a constraint programming model. We present an experimental study based on different levels of search intensification. The main contribution of this work is to propose a new matheuristic that reaches better quality solutions than before while decreasing the uncertainty of the makespan interval.

The paper is structured as follows: the problem is defined in Section II and the methodology is described in detail in Section III. In Section IV, the results of various experiments on instances from the literature are given. Finally, the conclusions and future work are presented in Section V.

Supported by the Spanish Government under research grant PID2019-106263RB-I00.

II. The Interval Job Shop Scheduling Problem

In the classical job shop problem, there is a set $J = \{J_1, \ldots, J_n\}$ of jobs to be executed on a set $M = \{M_1, \ldots, M_m\}$ of machines. Each job $J_j, j = 1, \ldots, n$, is divided into a set of tasks $m_j \leq m$. *Job precedence constraints* state that task belonging to a job J_j should be performed in a certain order, $(o(j,1), \ldots, o(j,m_j))$. *Resource constraints* ensure that each task $o(j,l)$ is executed on machine $\nu_{o(j,l)} \in M$ in an uninterrupted way. For the sake of simplicity, it is assumed that the tasks are numbered from 1 to $N = \sum_{j=1}^{n} m_j$ w.l.o.g., therefore a task $o(j,l)$ can be referred by its index $o = \sum_{i=1}^{j-1} m_i + l$ where $o \in O = \{1, \ldots, N\}$, and ν_o, p_o would represent its required machine and its processing time, respectively.

A solution to this problem is a *schedule* s, that is, an assignment of starting times s_o to each task o that is feasible with respect to all constraints and optimal according to a criterion. In our case, we target the most common optimization criterion, which is makespan minimization. Given the completion time of each task $c_o = s_o + p_o$, the makespan C_{max} is computed as $C_{max} = \max_{o \in O}\{c_o\}$. A schedule $\boldsymbol{s}$ sets an order π among tasks of jobs that require the same set of machines.

A. Interval Scheduling

In case of partial information about the duration of a task, if only an upper and a lower bound of the processing time of a task is known, then this uncertain duration can be naturally modeled as a closed interval of possible values denoted $\mathbf{a} = [\underline{a}, \overline{a}] = \{x \in \mathbb{R} : \underline{a} \leq x \leq \overline{a}\}$.

Let $\mathbb{IR}$ denote the set of closed intervals. To compute a solution to IJSP with makespan minimization, two arithmetic operations on $\mathbb{IR}$ are required: addition and maximum. These are defined as an extension of the corresponding operations on real numbers [20]. Given two intervals $\mathbf{a} = [\underline{a}, \overline{a}], \mathbf{b} = [\underline{b}, \overline{b}] \in \mathbb{IR}$:

$$\begin{aligned} \mathbf{a} + \mathbf{b} &= [\underline{a} + \underline{b}, \overline{a} + \overline{b}] \\ \max(\mathbf{a}, \mathbf{b}) &= [\max(\underline{a}, \underline{b}), \max(\overline{a}, \overline{b})] \end{aligned} \tag{1}$$

Besides the arithmetic operations, comparison is key in an optimization context, since there is no natural order between two intervals. A study provided in [4] introduce the concept of *admissible orders*, being those orderings in $\mathbb{IR}$ that hold certain desirable properties. Three of them are also introduced, namely $\leq_{Lex1}$, $\leq_{Lex2}$ and $\leq_{YX}$. In a scheduling problem, $\leq_{Lex1}$ and $\leq_{Lex2}$ could be seen as optimistic and pessimistic respectively, due to these orderings giving priority to the lower/upper bounds of the interval respectively. $\leq_{YX}$ provides a more balanced comparison, where the midpoint of the interval is the main factor, but the width of the interval is also considered:

$$\mathbf{a} \leq_{YX} \mathbf{b} \Leftrightarrow \underline{a} + \overline{a} < \underline{b} + \overline{b} \vee (\underline{a} + \overline{a} = \underline{b} + \overline{b} \wedge \overline{a} - \underline{a} \leq \overline{b} - \underline{b})$$

This ranking is an admissible version and arguably, a refinement of the ranking $\leq_{MP}$, defined as:

$$\mathbf{a} \leq_{MP} \mathbf{b} \Leftrightarrow E[\mathbf{a}] \leq E[\mathbf{b}]$$

where $E[\mathbf{a}]$ represents the Expected Value of the interval $\mathbf{a}$, $E[\mathbf{a}] = (\underline{a} + \overline{a})/2$, which coincides with the midpoint of the interval.

Since the starting and completion times of all tasks and the makespan are intervals, which can be interpreted as the possible values that the time they may take, a schedule $\boldsymbol{s}$ is an *interval-valued schedule*. Nevertheless, the task processing ordering π that determines the schedule is crisp, that is, there is no uncertainty concerning the processing order of tasks.

III. Methodology

In [12], a different ranking is used to compare the makespan intervals and establish the minimization criterion. Preliminary results for the IJSP are also given. In [5], it is established that the ranking used in [12] is actually equivalent to the $\leq_{MP}$ ordering. The authors also prove that the use of $\leq_{YX}$, despite focusing more on the interval width, does not obtain worse results than using $\leq_{MP}$ in terms of the expected makespan value. As claimed before, this is an expected result, since $\leq_{YX}$ can be seen as a refinement of using the midpoint. In this scenario, where both orderings offer similar expected value, we assume that a narrower interval of possible values is more preferable. This reduces the amount of uncertainty of the outcome of a real realization of the schedule, which in a sense, can be understood as having a more robust solution. In line with this idea, we propose to minimize the makespan using the $\leq_{YX}$ order.

We propose a matheuristic approach that combines a genetic algorithm (GA) with the resolution of a mathematical model. In our method, the GA uses the $\leq_{YX}$ order during the optimization process. The extensive literature in job shop problems show that optimizing makespan, even in the crisp case, is NP-hard and metaheuristics such as GA are among the most successful methods. However, the lexicographic nature of $\leq_{YX}$ may lead the genetic algorithm to find solutions that are good in terms of expected makespan, but with a lot of uncertainty. To find solutions with low uncertainty, the mathematical model will take solutions provided by GA and minimize the width of the makespan interval without increasing its expected value. Moreover, setting this constraint limits the size of the search space enough to help exact solvers find solutions quicker.

A. Mathematical Model

In this section, we present a mathematical model that minimizes the width of the makespan interval without increasing the expected makespan of an initial solution. In the scope of this work, the initial solution is provided by the GA and its makespan is represented by $\mathbf{C}^{\mathbf{in}}_{\max}$.

$$
\begin{aligned}
\min \quad & \bar{C}_{\max} - \underline{C}_{\max} \\
\text{s.t.} \quad & \bar{C}_{\max} + \underline{C}_{\max} \leq \underline{C}^{in}_{\max} + \bar{C}^{in}_{\max} && (2)\\
& \underline{c}_o = \underline{s}_o + \underline{p}_o \qquad \forall o \in O && (3)\\
& \bar{c}_o = \bar{s}_o + \bar{p}_o \qquad \forall o \in O && (4)\\
& \underline{C}_{\max} = \max\{\underline{c}_o\} \qquad \forall o \in O && (5)\\
& \bar{C}_{\max} = \max\{\bar{c}_o\} \qquad \forall o \in O && (6)\\
& \underline{s}_{o(j,l)} \geq \underline{c}_{o(j,l-1)} \qquad 1 \leq l \leq m_j, 1 \leq j \leq n && (7)\\
& \bar{s}_{o(j,l)} \geq \bar{c}_{o(j,l-1)} \qquad 1 \leq l \leq m_j, 1 \leq j \leq n && (8)\\
& \underline{s}_o \leq \underline{c}_{o'} \vee \underline{s}_o \geq \underline{c}_{o'} \quad \forall o \neq o' \in O : \nu_o = \nu_{o'} && (9)\\
& \bar{s}_o \leq \bar{c}_{o'} \vee \bar{s}_o \geq \bar{c}_{o'} \quad \forall o \neq o' \in O : \nu_o = \nu_{o'} && (10)\\
& \underline{s}_o \leq \bar{s}_o \qquad \forall o \in O && (11)
\end{aligned}
$$

The objective function minimizes the difference between the end points of the makespan interval. Constraints (2) ensure that the expected makespan of the new solutions is not larger than $E[\mathbf{C^{in}_{max}}]$. Constraints (3) and (4) define the completion time of a task whereas constraints (5) and (6) define the makespan as the maximum completion time of all tasks. The order among the tasks of the same job is provided by constraints (7) and (8). Constraints (11) guarantee the proper order between the end points of the starting time interval. Constraints (9) and (10) ensure that two or more tasks do not overlap on a machine.

In the context of this work, the mathematical model is adapted to a *constraint programming* (CP) model. In the CP formulation, end points of each machine are represented with a variable ($\underline{r}_m, \bar{r}_m$ where $m \in M$), then $\texttt{unary} - \texttt{resource}$ constraints are used for each end point of each machine. Also, $\texttt{same} - \texttt{sequence}$ constraints are used to ensure that the same processing order is followed for both $\underline{r}_m$ and $\bar{r}_m$.

B. The Matheuristic Algorithm

For the general scheme of out method, we shall adopt the structure of the genetic algorithm proposed in [5], which conforms the state of the art in IJSP with makespan minimization, to the best of our knowledge. The integration of the mathematical model into the general scheme can be done in several ways. A common option is to apply this kind of intensification immediately after the evaluation of the offspring population. This option has the advantage of avoiding to apply the intensification method twice to the same individual, in case the best individual does not change for several iterations. On the other hand, the newly improved individual can be potentially lost at the replacement stage if it does not manage to provide a significant improvement to the population. In our case, since the genetic part and the CP component focus on different parts of the makespan interval, it may happen that the solution given by the CP solver is not interesting enough in terms of midpoint to pass to the replacement phase, which uses $\leq_{YX}$ order to decide. This would waste the computational effort of solving the CP model and its contribution to the overall algorithm would be lost. It would be more beneficial to keep the solution in the population for next iterations, so that it can be recombined with other solutions. This may also help guiding the population towards solutions with a narrow interval width, besides having a good midpoint. Figure 1 illustrates the general schema of the proposed matheuristic.

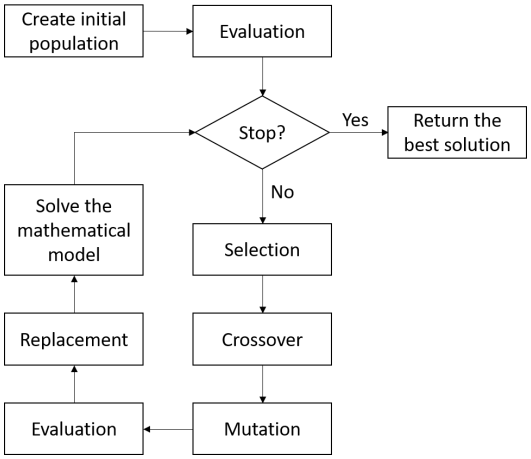


Fig. 1: General scheme of the matheuristic

The Crossover and Mutation blocks are applied each given a predetermined probability. In the figure, we assume that this check is a part of each respective block. Given the high computational cost that exact methods usually have, we opt to solve the mathematical model associated to the best solution of the population every $Iter_{CP}$ iterations, instead of at every iteration. This check is also part of the "Solve the mathematical model" block. To solve the mathematical model, the phenotype of the best solution found so far by the genetic algorithm, according to the $\leq_{YX}$ order, is given to the solver as initial solution. That is, the variables $\mathbf{s_o}$ of the model are set according to the solution given by the GA. Once the model returns an optimal solution, it is codified into a permutation following the encoding strategy given in [5] and then lamarckism is applied to feed the new solution to the population of the GA. In [5], the authors also provide the best setup for the GA, so the operators and probabilities of that component are taken from that work. That is, Job-Order Crossover operator with probability 1.0, insertion mutation with probability 0.15, selection by population shuffle, and a 2/4 tournament between parents and offspring as replacement strategy. The population size is 250 and the stopping criterion is 500 iterations.

IV. Experimental Results

In this section, we present the results of our solution method and assess the advantages of incorporating a mathematical model into a genetic algorithm. All the experiments have been executed on a PC with Intel Xeon Gold 6132 processor at 2.6 Ghz and 128 Gb RAM with Linux (CentOS v6.10), using a C++ implementation. The CP model is solved by IBM CP Optimizer version 12.9 with a time limit of 300 seconds using a single thread.

TABLE I: Average results of all instances

	10 calls			5 calls			2 calls			1 call			GA		
	MP	IW	Time	MP	IW	Time	MP	IW	Time	MP	IW	Time	MP	IW	Time
ABZ5	1278.0	88.0	51.8	1279.7	84.6	24.8	1282.5	79.0	11.6	1294.2	55.7	7.5	1294.2	90.1	1.9
ABZ6	987.5	79.0	18.4	987.6	78.7	9.9	987.6	78.9	5.2	990.6	72.9	4.0	990.6	80.2	1.9
LA16	992.5	91.0	14.2	992.5	91.0	9.7	994.3	87.3	5.6	1009.3	57.4	4.4	1009.3	95.1	2.2
LA17	826.5	75.0	10.8	826.5	75.0	7.8	827.0	73.9	4.5	834.1	59.9	3.7	834.1	83.9	1.9
LA18	898.4	84.1	18.9	898.6	83.1	10.2	898.5	83.5	5.3	900.9	78.7	3.7	900.9	80.6	1.8
LA19	889.5	87.0	63.5	888.8	88.4	37.3	888.7	88.7	16.8	892.6	80.9	9.5	892.6	89.2	1.9
LA20	949.8	82.5	23.6	950.4	81.1	11.6	950.5	81.0	6.4	952.4	77.3	4.3	952.4	79.3	1.9
LA21	1109.7	74.7	532.1	1118.3	57.5	344.7	1126.7	40.7	195.1	1130.1	33.8	114.7	1130.1	101.0	3.4
LA22	993.0	82.1	63.8	993.3	81.3	43.0	1001.4	65.1	24.3	1002.6	62.8	16.3	1002.6	104.4	3.5
LA23	1074.1	83.9	22.0	1074.1	83.8	9.5	1074.5	83.1	7.2	1074.8	82.5	5.8	1074.8	85.1	3.2
LA24	1005.8	49.8	168.4	1013.5	33.9	105.4	1017.1	27.2	71.7	1024.3	13.0	46.0	1024.5	94.3	3.2
LA25	1034.2	89.5	274.8	1039.0	80.0	176.1	1048.0	62.0	94.5	1047.7	62.6	56.4	1047.7	102.8	3.3
ORB01	1126.8	98.8	407.7	1126.8	97.7	220.0	1129.4	93.0	108.4	1140.4	70.4	68.3	1140.4	102.7	2.2
ORB02	930.0	82.0	34.8	930.0	82.0	22.5	930.7	80.7	11.7	956.9	28.3	8.3	956.9	82.7	2.1
ORB03	1069.0	98.0	157.0	1069.0	98.0	92.8	1072.4	91.1	52.0	1090.9	54.3	35.7	1090.9	107.3	2.2
ORB04	1068.1	95.9	97.2	1068.8	94.5	57.1	1071.1	89.8	25.0	1078.3	75.5	15.3	1078.3	94.4	2.3
ORB05	947.4	83.5	59.9	947.3	84.0	37.8	947.6	83.3	17.5	948.0	82.6	11.3	948.0	104.0	2.2
Average	*1010.6*	*83.8*	*118.8*	*1012.0*	*80.9*	*71.8*	*1014.6*	*75.8*	*39.0*	*1021.6*	*61.7*	*24.4*	*1021.6*	*92.8*	*2.4*

For testing the matheuristic approach, we make use of the same 17 instances that appear in [5]: ABZ5–6 (10×10), La16–20 (10×10), La21–25 (15×10), ORB01–05 (10×10).

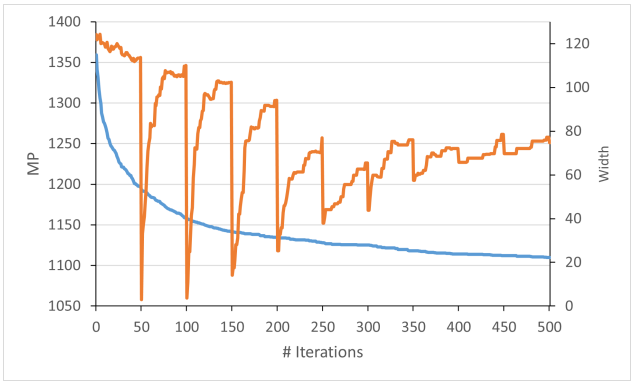
The algorithm is run 30 times with the stopping criterion of 500 iterations. For the purpose of comparison, GA is also run alone on the same instances under the same conditions. To explore the impact of integrating the mathematical model, four experimental setups are used: calling the CP solver once every 50 iterations (10 calls in total) for the incumbent solution, once every 100 iterations (5 calls), once every 250 iterations (2 calls) and finally only once at the end of the evolutionary process (1 call). At each call, the CP solver is given a time limit of 300 seconds.

The average results of tested configurations of the matheuristic and the GA over 30 runs are given in Table I. The midpoint (MP) and width of the makespan interval (IW) are presented along with average runtimes in seconds (Time). Averages of all columns are given in the last line. It can be observed that on average, the matheuristic takes much longer time than the GA in all setups, which is expected. It is also indicative that the majority of the runtime consumed by the matheuristic is used by the solver. However, despite being given 300 seconds for each call, the solver does never use up this time. In other words, it finds the optimal makespan interval width without worsening the midpoint much sooner than hitting the time limit for every call and instance.

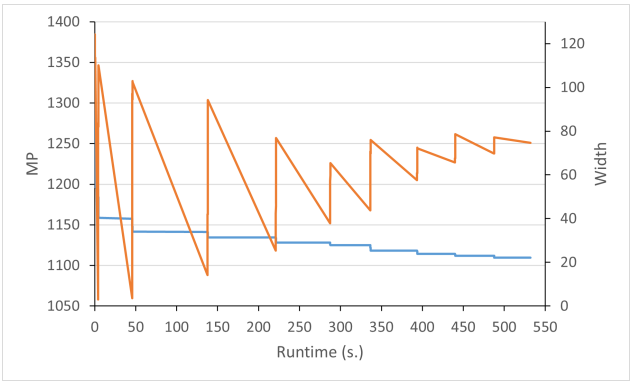
One may argue that if the GA was given much more time, it would also result in much better solutions. To test if that is the case, the GA is launched given the same amount of time as the matheuristic with 10 calls as stopping criterion for all instances. We shall refer to the original GA as GA_{iter} and to the GA with the new stopping criterion as GA_{time}. The results obtained when given more time to GA show an average improvement of 0.2% in the midpoint of the makespan interval. In the same amount of time, the improvement that the matheuristic with 10 calls obtains w.r.t. GA_{iter} is 1.1%, that is four times the improvement obtained by simply giving GA more time. Moreover the average interval width obtained by the matheuristic is 9.0% better than GA, while giving GA more time actually increases the width in 0.2% due to its main focus on optimizing the midpoint. In instance La24, considered to be a very challenging instance for job shop ([2]), the matheuristic improves the interval midpoint obtained by GA in 1.8% while reducing the interval width in 47%.

Reducing the makespan interval width when using the matheuristic is an expected outcome, but what is more interesting is the reduction in the interval midpoint. To better understand the reason behind this, a comparison of the matheuristic with 10 calls and the GA can be found in Figure 2 for instance La21. In Figures 2a and 2c, the makespan midpoint and width values are plotted with respect to the number of iterations of the genetic component and the stopping criterion is the number of iterations. In Figures 2b and 2d, these values are plotted with respect to run time in seconds. In this case, GA is allowed to run for as much time as the matheuristic takes to meet the stopping criterion. It is obvious that the GA had already converged at iteration 500 and increasing its run time does not improve its outcome, which is the case for other instances as well. On the other hand, the descending parts of the orange lines (interval width) actually show the time that the solver is using to improve the makespan width, while the ascending parts correspond to the GA, which focuses on reducing the midpoint.

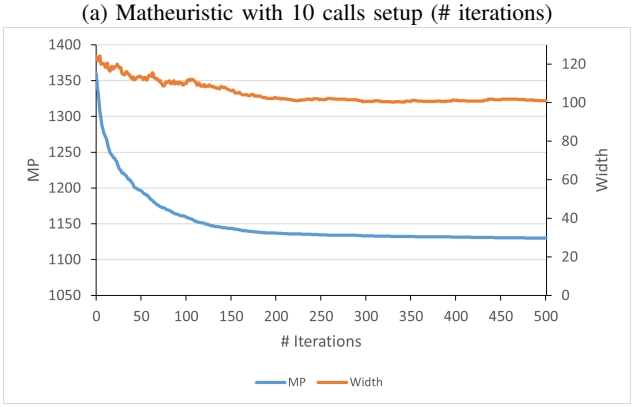
It can be observed that there is a significant drop in the makespan interval width every 50 iterations in Figure 2a, or in other words, every time the solver is called with the incumbent solution as an input. This behavior is expected since the objective of the mathematical model is to minimize the makespan width without deteriorating the midpoint. It is also observed that when the midpoint values decrease, the



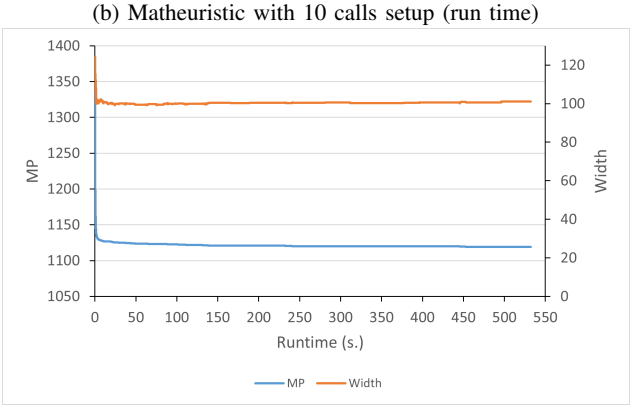
(a) Matheuristic with 10 calls setup (# iterations)



(b) Matheuristic with 10 calls setup (run time)



(c) GA_iter



(d) GA_time

Fig. 2: Comparison of matheuristic and GA over the instance La21

width tends to increase, or at least it ceases to improve. However, after the solver call, an interesting synergy emerges: the evolutionary algorithm takes this solution with smaller interval width to generate schedules with smaller midpoints. This phenomenon can be observed more clearly in Figure 2b: immediately after every call to the solver, a drop in the midpoint can be observed. The reason for this is the following: let the incumbent makespan interval that the GA finds before calling the solver be called **a**. When the solver is given this solution, since the model aims to decrease the width but it cannot surpass the given midpoint, typically it returns a solution with makespan **b** where $\overline{b} \leq \overline{a}$ and $\underline{b} \geq \underline{a}$. To find a better $\overline{b}$ value, it tends to rearrange the task order of the solution, while it can easily increase the value of $\underline{b}$ by delaying the starting times of some tasks. When this solution returns to GA, which focuses on the midpoint, it reverts the delay induced by the solver, obtaining a new makespan value **c** where $\overline{c} = \overline{b}$ but $\underline{c} \leq \underline{b}$. In other words, the GA takes advantage of the task rearrangements found by the solver to then decrease the makespan interval midpoint. Although the model only minimizes the width, in return it helps the GA to minimize the midpoint.

When the CP model is called 5 times instead of 10, the improvement in the midpoint and width of the makespan interval with respect to the GA solution, is 0.9% and 12.1%. These values become 0.7% and 17.1% respectively when the solver is called twice and, 0% and 32.5% when the solver is called once. The impact of incorporating the mathematical model is visually represented in Figure 3 for instance La21. The behavior is similar for the rest of instances. The midpoint and width values of the makespan interval for the matheuristic with all setups, the GA with stopping criteria of number of iterations (GA_iter) and run time (GA_time) are plotted. As expected, when the solver is applied only at the last iterations of the matheuristic, the obtained effect is "horizontal", i.e. the expected value is the same, since it was set as a constraint, but the interval width decreases. One would expect that by applying the solver more often, the obtained solution would be somehow in a straight line with the solutions obtained by the GA and the setup with 1 call to the solver. However, the results show a different effect: increasing the number of calls to the solver leads to solutions that actually have a better midpoint than the GA, and an interval width between the one obtained by GA and by one call to the solver. In fact, by varying the frequency in which we solve the model, we obtain what we would call a set of non-dominated solutions if this was a multi-objective problem. Moreover, it is more obvious that when GA is given the same time as the matheuristic with 10 calls to the solver, it cannot surpass the midpoint obtained with 5 calls besides having a much larger makespan width. This

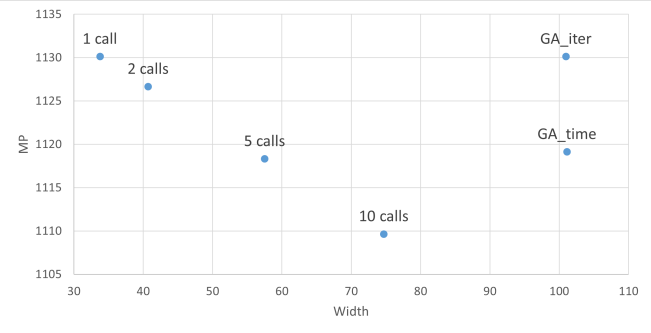


Fig. 3: Depiction of the synergy between the GA and the mathematical model over instance La21

goes in accordance with the graphics seen in Figure 2, where larger improvements of the interval midpoint would also lead to having less room to find narrow intervals.

V. Conclusion

In this work, we tackle a job shop scheduling problem variant with interval durations. The objective is to minimize makespan via a ranking method that minimizes the midpoint and width of the makespan interval. We propose a novel matheuristic method that integrates a mathematical model which is solved by a commercial solver into a state-of-the-art genetic algorithm. The GA focuses on minimizing the makespan with respect to $\leq_{YX}$ ordering, whereas the model aims to minimize only the makespan width. This approach is tested on 17 instances from literature with different frequency of calls to the exact solver to analyze the synergy between these two methods.

The results suggest that the matheuristic is capable of providing better quality makespan intervals than the GA in terms of both midpoint and width. Although the matheuristic algorithm has a greater run time, it is shown that the GA cannot obtain similar results when it is given the same amount of time. Furthermore, the more times the mathematical model is solved, the smaller is the midpoint, which is a counter-intuitive outcome but shows the great synergy effect obtained with the method.

Future extensions of this work could follow various paths. First step would be modifying the algorithm in terms of stopping criterion and call conditions of the mathematical model to have a more dynamic and adaptive approach. Later, a more sophisticated metaheuristic can be used in place of the genetic algorithm to further improve the solution quality. Applying this approach to JSP variants with uncertainty, such as flexible JSP, would also be a worthwhile direction for future research.

References

[1] A. Allahverdi, H. Aydilek, and A. Aydilek. Single machine scheduling problem with interval processing times to minimize mean weighted completion time. *Computers & Operations Research*, 51:200–207, 2014.

[2] D. Applegate and W. Cook. A computational study of the job-shop scheduling problem. *ORSA Journal of Computing*, 3:149–156, 1991.

[3] M. A. Boschetti, V. Maniezzo, M. Roffilli, and A. B. Röhler. Matheuristics: Optimization, simulation and control. In *International Workshop on Hybrid Metaheuristics*, pages 171–177. Springer, 2009.

[4] H. Bustince, J. Fernandez, A. Kolesárová, and R. Mesiar. Generation of linear orders for intervals by means of aggregation functions. *Fuzzy Sets and Systems*, 220:69–77, 2013.

[5] H. Díaz, I. González-Rodríguez, J. J. Palacios, I. Díaz, and C. R. Vela. A Genetic Approach to the Job Shop Scheduling Problem with Interval Uncertainty. In *Communications in Computer and Information Science*, volume 1238 CCIS, pages 663–676. 2020.

[6] H. Díaz, J. J. Palacios, I. Díaz, C. R. Vela, and I. González-Rodríguez. Tardiness minimisation for job shop scheduling with interval uncertainty. In *Hybrid Artificial Intelligent Systems*, pages 209–220. Springer, 2020.

[7] D. Dubois, H. Prade, and S. Sandri. On possibility/probability transformations. In *Fuzzy Logic*, volume 12 of *Theory and Decision Library*, pages 103–112. Kluwer Academic, 1993.

[8] L. Fanjul-Peyro, F. Perea, and R. Ruiz. Models and matheuristics for the unrelated parallel machine scheduling problem with additional resources. *European Journal of Operational Research*, 260(2):482–493, 2017.

[9] M. A. González, A. Oddi, and R. Rasconi. Multi-objective optimization in a job shop with energy costs through hybrid evolutionary techniques. In *Proceedings of the 27th International Conference on Automated Planning and Scheduling (ICAPS-2017)*, pages 140–148, 2017.

[10] Y. Han, D. Gong, J. Yaochu, and Q.-k. Pan. Evolutionary multi-objective blocking lot-streaming flow shopscheduling with interval processing time. *Applied Soft Computing*, 42:229–245, 2016.

[11] H. Ishibuchi, N. Tsukamoto, and Y. Nojima. Evolutionary many-objective optimization: A short review. In *Genetic and Evolving Systems, 2008. GEFS 2008. 3rd International Workshop on*, pages 47–52, 2008.

[12] C. Jiang, X. Han, G. Liu, and G. Liu. A nonlinear interval number programming method for uncertain optimization problems. *European Journal of Operational Research*, 188(1):1–13, 2008.

[13] D. Lei. Population-based neighborhood search for job shop scheduling with interval processing time. *Computers & Industrial Engineering*, 61:1200–1208, 2011.

[14] D. Lei. Interval job shop scheduling problems. *International Journal of Advanced Manufacturing Technology*, 60:291–301, 2012.

[15] D. Lei. Multi-objective artificial bee colony for interval job shop scheduling with flexible maintenance. *The International Journal of Advanced Manufacturing Technology*, 66(9-12):1835–1843, 2013.

[16] D. Lei and S. Cao. A novel shuffled frog-leaping algorithm for flexible job shop scheduling with interval processing time. In *Proceedings of the 36th Chinese Control Conference*, pages 2708–2713, 2017.

[17] D. Lei and X. Guo. An effective neighborhood search for scheduling in dual-resource constrained interval job shop with environmental objective. *International Journal of Production Economics*, 159:296–303, 2015.

[18] S.-W. Lin and K.-C. Ying. Optimization of makespan for no-wait flowshop scheduling problems using efficient matheuristics. *Omega*, 64:115–125, 2016.

[19] N. M. Matsveichuk, Y. N. Sotskov, N. G. Egorova, and T.-C. Lai. Schedule execution for two-machine flow-shop with interval processing times. *Mathematical and Computer Modelling*, 49(5-6):991–1011, 2009.

[20] R. E. Moore, R. B. Kearfott, and M. J. Cloud. *Introduction to Interval Analysis*. Society for Industrial and Applied Mathematics, 2009.

[21] S. Petrovic, S. Fayad, D. Petrovic, E. Burke, and G. Kendall. Fuzzy job shop scheduling with lot-sizing. *Annals of Operations Research*, 159:275–292, 2008.

[22] M. L. Pinedo. *Scheduling. Theory, Algorithms, and Systems.* Springer, fifth edition, 2016.

[23] Y. N. Sotskov and N. G. Egorova. Single machine scheduling problem with interval processing times and total completion time objective. *Algorithms*, 11(5):66, 2018.

[24] S. Voss, V. Maniezzo, and T. Stützle. Matheuristics: Hybridizing metaheuristics and mathematical programming (annals of information systems). 2009.

Solving the Permutation Heijunka Flow Shop Scheduling Problem with Non-unit Demands for Jobs*

*Note: The full contents of this paper have been published in the volume Lecture Notes in Artificial Intelligence 12882 (LNAI 12882)

Joaquín Bautista-Valhondo
IOC ETSEIB
Universitat Politcnica de Catalunya
Barcelona, Spain
joaquin.bautista@upc.edu

***Abstract*—In this paper a problem of job sequences in a workshop is presented, taking into account non-unit demands for these and whose objective is to obtain manufacturing sequences satisfying the Quota property with a minimum total completion time for all the jobs (C_{max}). Two procedures are proposed to solve the problem: Mixed Integer Linear Programming and a Metaheuristic based on Multistart and Local Search. The two proposed procedures are tested using instance set Nissan-9Eng.I, in both cases giving rise to highly satisfactory performance both in quality of solutions obtained and in the CPU times required.**

***Index Terms*—Flow Shop Scheduling Problem, Overall Demand, Heijunka, Mixed Integer Linear Programming, Multistart, Local Search**

DOI: https://doi.org/10.1007/978-3-030-85713-4_17

Combinando GRASP con *Iterated Greedy* para el problema de la máxima intersección de *k*-conjuntos.

Alejandra Casado-Ceballos
Dept. Computer Science
Universidad Rey Juan Carlos
Móstoles, España
a.casadoce.2017@alumnos.urjc.es
0000-0003-3417-6859

Sergio Pérez-Peló
Dept. Computer Science
Universidad Rey Juan Carlos
Móstoles, España
sergio.perez.pelo@urjc.es
0000-0002-1915-4160

Jesús Sánchez-Oro
Dept. Computer Science
Universidad Rey Juan Carlos
Móstoles, España
jesus.sanchezoro@urjc.es
0000-0003-1702-4941

Abraham Duarte
Dept. Computer Science
Universidad Rey Juan Carlos
Móstoles, España
abraham.duarte@urjc.es
0000-0002-4532-3124

***Resumen*—El problema de la selección de individuos de una población en función de las características que comparten ha suscitado el interés de la comunidad científica en los últimos años. En este artículo se propone un algoritmo basado en la metaheurística *Iterated Greedy* para resolver el problema de la máxima intersección de k-conjuntos (kMIS). En concreto, se trata de una variante donde se deben seleccionar k individuos de una población, con el objetivo de maximizar el número de características que dichos individuos tienen en común. Para aumentar la eficiencia del algoritmo propuesto, se plantea una nueva representación de la solución basada en bitsets. Esta representación reduce notablemente la complejidad de la evaluación de la función objetivo. La propuesta algorítmica se compara con el mejor trabajo encontrado en la literatura reciente. Los resultados computacionales muestran que el algoritmo propuesto ofrece un rendimiento muy superior al previo tanto en calidad como en tiempo de cómputo, emergiendo como uno de los algoritmos más competitivos en el contexto del kMIS.**

***Index Terms*—kMIS, *GRASP*, *Iterated Greedy*, metaheurísticas**

I. Introducción

La selección de individuos de una población siguiendo diferentes criterios es uno de los problemas de optimización más extendidos. Entre ellos, destaca una familia de problemas cuyo cometido es seleccionar un conjunto de individuos de manera que se maximicen las características que estos tienen en común. Esta familia de problemas ha resultado en una gran variedad de problemas de optimización combinatoria, donde el objetivo suele consistir en maximizar las relaciones entre dos conjuntos diferentes de elementos, sin considerar las relaciones entre los elementos del mismo conjunto.

Uno de los problemas más estudiados de este tipo es el *Maximum edge biclique (MEB)* [1], donde el objetivo es encontrar un subgrafo bipartito completo (*biclique*) con el máximo número de aristas dado un grafo bipartito cualquiera. Este problema ha sido estudiado desde distintas perspectivas, usando tanto aproximaciones heurísticas como exactas [2], [3]. Además, existen numerosas variantes de este problema, considerando nuevas restricciones como el equilibrio entre conjuntos [4] o los pesos de los elementos para seleccionarlos según su relevancia [2].

Esta investigación está financiada por el Ministerio de Ciencia, Innovación y Universidades con el proyecto con referencia PGC2018-095322-B-C22, y por la Comunidad de Madrid y los Fondos Estructurales de la Unión Europea con los proyectos con referencias S2018/TCS-4566 y Y2018/EMT-5062.

Esta investigación está centrada en el problema que maximiza la intersección entre k elementos (kMIS), y es muy similar a la familia de problemas del MEB. Dado un conjunto de elementos $E = \{e_1, e_2, \ldots, e_n\}$, donde cada e tiene un conjunto de características $F_e \subseteq F$, siendo F el conjunto de las características posibles. Siendo así, dado un número entero k, el objetivo de kMIS es seleccionar un subconjunto de elementos de E que tengan el máximo número de características de F en común.

Una solución factible para el problema kMIS está formada por un conjunto de k elementos de E. Dada una solución $S \subseteq E$, donde $|S| = k$, el valor de la función objetivo $k\text{MIS}(S)$ se evalúa como:

$$k\text{MIS}(S) = \left| \bigcap_{e \in S} F_e \right|$$

El objetivo del kMIS es encontrar una solución $S^\star$ con el máximo valor de $k\text{MIS}(S^\star)$. Más formalmente se podría definir como

$$S^\star = \arg\max_{S \in \mathbb{S}} k\text{MIS}(S)$$

donde $\mathbb{S}$ es el conjunto completo de soluciones que se pueden explorar, es decir, el espacio de búsqueda.

La figura 1(a) muestra una instancia del problema kMIS compuesta por 4 elementos y 5 posibles características con las que ser relacionados. Las características de cada elemento están definidas por una línea entre ellos. Para facilitar su identificación, el color de las líneas se corresponde con el del elemento al que pertenece la relación. Por ejemplo, las características asociadas al elemento e_1 son $F_{e_1} = \{f_1, f_2, f_3\}$, las de e_2 son $F_{e_2} = \{f_1, f_2, f_3, f_5\}$, y así con cada elemento.

Para esta instancia, se han calculado dos posibles soluciones S_1 y S_2. Por un lado, S_1, representada en la figura 1(b), está formada por los elementos $S_1 = \{e_1, e_3, e_4\}$ y el valor de la función objetivo es $k\text{MIS}(S_1) = |F_{e_1} \cap F_{e_3} \cap F_{e_4}| = 1$. Por otro lado, S_2, que se muestra en la figura 1(c), contiene los elementos $S_2 = \{e_1, e_2, e_3\}$ y tiene como valor de la función objetivo $k\text{MIS}(S_2) = |F_{e_1} \cap F_{e_2} \cap F_{e_3}| = 3$. Es por esto que,

sabiendo que tratamos de maximizar la función objetivo, está claro que S_2 es mejor que S_1.

Este problema tiene múltiples aplicaciones en distintas áreas en la vida real. Los autores de [5] muestran que el kMIS es útil para controlar la privacidad de ciertos datos. Además, en el contexto de la genética [6], el objetivo es seleccionar k genes con el máximo número de características en común. [7] propone una aplicación donde cada elemento es un artista musical y cada característica una persona. Ambos tienen relación en caso de que la persona sea admiradora del músico. El objetivo es encontrar el conjunto de k músicos maximizando los admiradores en común para organizar un festival.

El kMIS fue definido por primera vez por [5], y se trata de un problema $\mathcal{NP}$-difícil como se demostró en [8]. Aun así, para ciertas instancias especiales, el kMIS puede ser resuelto de forma óptima en tiempo polinomial. [9] propuso un algoritmo con complejidad polinomial que enumeraba todos los conjuntos cercanos de una solución factible representando la instancia como un grafo bipartito y mostrando todos los bicliques máximos del grafo si el valor de k está limitado por una constante.

[7] propuso un método exacto basado en un modelo de programación entera para el kMIS que comenzaba la búsqueda partiendo de una solución de buena calidad obtenida a través de un método de preprocesamiento rápido. Sin embargo, como un método exacto, requiere mucho tiempo de computación para solucionar de forma óptima instancias incluso de pequeño tamaño. Adicionalmente proponen tres nuevos modelos de programación entera lineal y un método de construcción heurística para el kMIS, donde dos de las fórmulas presentadas fueron adaptadas del problema MEB [10].

A pesar de la dificultad asociada a encontrar soluciones óptimas para el kMIS, el problema ha sido ignorado desde un punto de vista heurístico. Actualmente el mejor trabajo previo que utiliza un enfoque heurístico, [11] propone la combinación de distintos algoritmos heurísticos y metaheurísticos para proporcionar soluciones de alta calidad en tiempos de computación reducidos. La propuesta está basada en *Variable Neighborhood Search (VNS)*, propuesto por [12], con el objetivo de aprovechar los cambios de vecindad para mejorar las soluciones encontradas durante la búsqueda. En concreto, amplían el procedimiento constructivo propuesto en [7] para tratar varias soluciones simultáneamente combinándolas mediante Path Relinking. Además, proponen un *VNS* reactivo que modifica la fase de perturbación incluyendo una selección voraz aleatorizada de soluciones para ser seleccionadas. Finalmente, el método utiliza el cambio de vecindad para escapar de óptimos locales en caso de estancarse.

Este artículo está estructurado de la siguiente manera: la sección II describe la representación de la solución usada en esta investigación, la sección III presenta la propuesta algorítmica basada en *GRASP*, la sección IV describe el algoritmo basado en *Iterated greedy* utilizado como post-proceso para mejorar las soluciones encontradas por *GRASP*, la sección V muestra la experimentación realizada para analizar el comportamiento del algoritmo propuesto y, finalmente, la sección VI resume una serie de conclusiones derivadas de esta investigación.

II. Representación de la solución

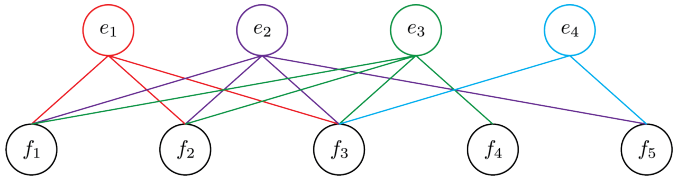
Dada una instancia $I = (B, k)$, donde $B = (E, F)$ es un grafo bipartito formado por los conjuntos de vértices E y F, una implementación directa para el kMIS consideraría que la solución se representa por un conjunto de elementos seleccionados de E. Sin embargo, es importante tener en cuenta el rendimiento de las operaciones más comunes sobre esta representación de la solución para reducir el tiempo de cómputo. Si consideramos esta representación, la evaluación de la solución se calcula mediante la intersección de las características comunes de los elementos seleccionados, que puede ser computacionalmente exigente para instancias medianas y grandes. Con el objetivo de acelerar el proceso, proponemos el uso de bitsets como adaptación a la idea original de bit-strings presentada en [11].

Un bitset se define como un array de elementos donde el valor de cada uno solo puede ser 0 o 1. Para modelar el kMIS, cada elemento $e \in E$ tiene su propio bitset B_e de tamaño $|F|$, donde el valor 1 en una posición p indica que F_e tiene la característica f_p. Además, para evaluar las características en común entre dos elementos e_i y e_j, solo es necesario realizar la operación lógica `AND` entre B_i y B_j, que se ejecuta en tiempo constante. Este comportamiento nos permite acelerar el proceso de búsqueda. La tabla I muestra la representación del bitset y la evaluación de la función objetivo de las soluciones S_1 y S_2 que aparecen en la figura 1.

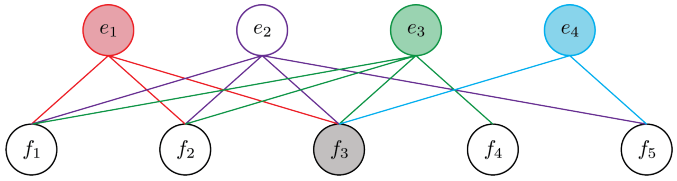
	S_1					S_2				
	f_1	f_2	f_3	f_4	f_5	f_1	f_2	f_3	f_4	f_5
e_1	1	1	1	0	0	1	1	1	0	0
e_2	1	1	1	0	1	1	1	1	0	0
e_3	1	1	1	1	0	1	1	1	0	0
e_4	0	0	1	0	1	0	0	1	0	1
AND	0	0	1	0	0	1	1	1	0	0

Tabla I: Representación de la solución S_2 con un bitset mostrada en la figura 1.

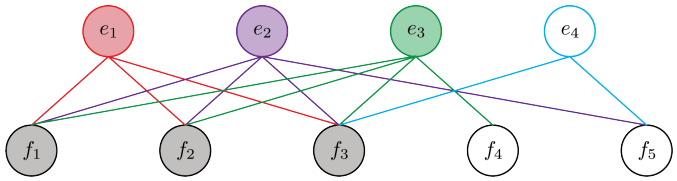
Con respecto a la representación de la solución, el elemento e_1 tiene las características f_1, f_2 y f_3, que están representadas con un 1 en la tabla. Las características f_4 y f_5 tienen un 0 en la tabla debido a que no son características que pertenezcan a e_1. Podemos hacer el mismo análisis con los elementos e_2, e_3 y e_4. Las filas resaltadas en gris indican los elementos seleccionados en cada solución. La última fila representa el resultado de realizar la operación lógica `AND` entre los bitsets de los elementos seleccionados. Para la solución S_1 el único bit activo tras la operación corresponde a la característica f_3, lo que supone que es la única característica que e_1, e_3, y e_4 tienen en común. En el caso de S_2, las características en común son f_1, f_2, y f_3, es por ello que en el resultado de realizar la operación lógica `AND` encontramos los bits de estas tres características activos. Para calcular el valor de la función objetivo a partir de esta información solo se tiene que evaluar la cardinalidad del bitset resultante, que viene dada por el número de bits activos. En el caso de S_1 el valor de la



(a) Instancia compuesta de 4 elementos y 5 características.



(b) Solución S_1



(c) Solución S_2

Figura 1: Instancia compuesta de 4 elementos y 5 características, 1(a), y dos soluciones factibles para esta instancia: S_1, 1(b), y S_2, 1(c).

función objetivo sería 1, mientras que S_2 tiene un valor de 3, siendo esta última mejor ya que se trata de un problema de maximización.

En términos de complejidad computacional, esta operación se realiza en tiempo constante, resultando en una complejidad de $O(1)$. Esta complejidad es menor que la que requiere calcular la intersección de características de los elementos seleccionados, ya que tiene una complejidad de $O(n)$.

III. *Greedy Randomized Adaptive Search Procedure*

Este trabajo propone un algoritmo basado en la metodología *Greedy Randomized Adaptive Search Procedure (GRASP)*. Se trata de una metaheurística multiarranque presentada en [13], que no fue definida formalmente hasta [14]. Está formada por dos fases: construcción y búsqueda local. La primera fase intenta generar una solución diversa y de alta calidad desde cero [15]. Por otro lado, la segunda fase es la responsable de encontrar un óptimo local respecto a cierta vecindad. La idea principal de *GRASP* recae en añadir aleatoriedad durante la fase de construcción, de forma que se diversifica la búsqueda. Al contrario que en una construcción plenamente voraz, cada construcción de *GRASP* genera una solución diferente, explorando una gran parte del espacio de búsqueda.

III-A. Constructivo

En este trabajo se propone un procedimiento constructivo que aprovecha la representación de la solución descrita en la sección II con el objetivo de generar soluciones prometedoras de las cuales partirá la búsqueda local. El procedimiento constructivo sigue el esquema tradicional del constructivo en *GRASP*. El algoritmo 1 muestra el pseudocódigo del constructivo propuesto.

Algoritmo 1 `Constructivo`$(B = (E, F), k, \alpha)$

1: $e \leftarrow Random(E)$
2: $S \leftarrow \{e\}$
3: $CL \leftarrow E \setminus \{e\}$
4: **while** $|S| < k$ **do**
5: $\quad g_{\text{mín}} \leftarrow \text{mín}_{c \in CL}\, g(c)$
6: $\quad g_{\text{máx}} \leftarrow \text{máx}_{c \in CL}\, g(c)$
7: $\quad \mu \leftarrow g_{\text{máx}} - \alpha \cdot (g_{\text{máx}} - g_{\text{mín}})$
8: $\quad RCL \leftarrow \{c \in CL : g(c) \geq \mu\}$
9: $\quad e \leftarrow Random(RCL)$
10: $\quad S \leftarrow S \cup \{e\}$
11: $\quad CL \leftarrow CL \setminus \{e\}$
12: **end while**
13: **return** S

Como suele ocurrir en *GRASP*, el primer elemento es seleccionado de forma aleatoria, favoreciendo la diversidad en las soluciones (paso 1). Tras ello, el elemento se incluye en la solución (paso 2) y la lista de candidatos *CL* se construye con todos los elementos de E menos el primer elemento seleccionado (paso 3). Iterativamente, el método añade elementos a la solución usando el criterio voraz elegido que se describirá posteriormente (pasos 4-12).

En cada iteración son evaluados el mínimo ($g_{\text{mín}}$) y el máximo ($g_{\text{máx}}$) valor de la función voraz, considerados para

guiar la construcción eligiendo entre los candidatos (pasos 5-6). Es recomendable usar una función voraz que permita calcular el valor de esta con un tiempo computacional bajo, ya que se debe evaluar en cada iteración. En el contexto de kMIS, la representación eficiente de la solución presentada en la sección II permite usar directamente la función objetivo como función voraz. El valor de la función voraz para un elemento c, denotado como $g(c)$, se calcula como el número de características que c y los elementos ya en la solución tienen en común. En términos de la representación de la solución, el valor de la función voraz se obtiene realizando la operación lógica `AND` entre las características de los elementos de la solución parcial y las características de c, y posteriormente calculando la cardinalidad del bitset resultante.

El umbral μ se calcula para determinar qué elementos deben entrar en la lista de candidatos restringida *RCL* (pasos 7-8). El umbral depende de un parámetro de entrada α, que puede tener valores en el rango entre 0 y 1. Es importante destacar que en el caso de que $\alpha = 0$ el valor de $\mu = g_{\text{máx}}$. Es por esto que solo los elementos con un valor de la función voraz igual a $g_{\text{máx}}$ son contemplados en la *RCL*, resultando en un algoritmo totalmente voraz. Por otro lado, en caso de que $\alpha = 1$, μ es igual al valor de $g_{\text{mín}}$, por lo que todos los elementos se incluyen en la *RCL* resultando un procedimiento totalmente aleatorio. Por lo tanto, cuanto mayor sea el valor de α más voraz será el algoritmo. En la sección V se analiza el valor más apropiado para este parámetro.

El siguiente elemento que se añade en la solución es seleccionado de forma aleatoria de la *RCL* (paso 9). Finalmente, la *CL* es actualizada eliminando el elemento seleccionado de forma que no se pueda elegir dos veces el mismo elemento. El método termina cuando son seleccionados k elementos de E, devolviendo la solución construida (paso 13).

III-B. Búsqueda local

La segunda fase de *GRASP* consiste en encontrar un óptimo local respecto a cierta vecindad empezando por la solución generada en el constructivo. La literatura de *GRASP* ha considerado tanto búsquedas locales simples como otras más complejas. Algunos trabajos han llegado a incluir metahurísticas completas usando *Variable Neighborhood Search* [16] o incluso algoritmos genéticos [17] en la segunda fase de *GRASP* con el objetivo de explorar más allá en el espacio de búsqueda.

En este trabajo se propone una simple pero efectiva búsqueda local, con el objetivo de encontrar un óptimo local en un tiempo computacional reducido. El primer elemento que se debe identificar en una búsqueda local es el movimiento que se aplica en la búsqueda. Debido a que una solución para el kMIS está formada por exactamente k elementos y con el objetivo de mantener la factibilidad de las soluciones exploradas en el vecindario, se propone un movimiento de intercambio. Dado un elemento $e_i \in S$ y un elemento $e_j \in E \setminus S$, se intercambia e_i con e_j. En términos matemáticos,

$$\textit{Intercambio}(S, e_i, e_j) = (S \setminus e_i) \cup e_j$$

La definición de un movimiento nos permite establecer la vecindad $N(S)$ que se explora en la búsqueda local propuesta como un conjunto de soluciones que se pueden generar a partir de S realizando un único movimiento de intercambio. Más formalmente,

$$N(S) = \{S' \leftarrow \textit{Intercambio}(S, e_i, e_j),\ \forall e_i \in S \wedge \forall e_j \in E \setminus S\}$$

Habiendo definido el movimiento y la vecindad generada a partir de este, el último elemento que falta por definir es la manera en la que se recorre esta vecindad. Tradicionalmente se consideran dos formas de recorrer una vecindad: *best improvement* y *first improvement*. Por un lado *best improvement* explora la vecindad de una solución al completo, realizando aquel movimiento que genera una solución de mayor calidad. Por otro lado, *first improvement* realiza el primer movimiento que mejora la solución de partida.

Debido a que *best improvement* requiere una exploración completa de la vecindad en cada iteración, suele ser computacionalmente más lenta que *first improvement*. En kMIS, elegimos *first improvement* con el objetivo de mantener la rapidez del algoritmo al incluir la búsqueda local. Al elegir *first improvement* el orden en el que las soluciones son exploradas es importante por lo que, para evitar sesgar la búsqueda, se ordenan de forma aleatoria en cada iteración. El algoritmo 2 muestra el pseudocódigo de la búsqueda local propuesta.

Algoritmo 2 `Búsqueda_local`$(I = (E, F, k), S)$

```
1:  best ← kMIS(S)
2:  for e_i ∈ S do
3:      for e_j ∈ E \ S do
4:          S ← Intercambio(S, e_i, e_j)
5:          if kMIS(S) > best then                ▷ Mejora
6:              go to 1
7:          end if
8:          S ← Intercambio(S, e_j, e_i)
9:      end for
10: end for
11: return S
```

La función comienza recorriendo los elementos de S de forma aleatoria (paso 2). Tras esto, para cada elemento $e_i \in S$ se realiza un movimiento de intercambio con cada elemento $e_j \in E \setminus S$ (paso 4). En caso de encontrar una mejora (paso 5) se reinicia la búsqueda (paso 6). Si no, el movimiento de intercambio se deshace, volviendo a la solución inicial. El procedimiento termina cuando no se encuentra ninguna mejora, devolviendo un óptimo local de la solución inicial S (paso 11).

La búsqueda local puede mejorar considerablemente ya que realizar un intercambio implica revaluar la solución en cada iteración. Para reducir el tiempo empleado en la búsqueda local, proponemos una búsqueda local más eficiente. Esta nueva búsqueda local es capaz de decidir si un intercambio generará una mejor solución o no realizando una única operación lógica `AND`. Para ello, el paso 4 se reemplaza por la operación `AND`

entre el bitset de la solución, cuando el elemento e_i ha sido borrado, y el bitset del elemento e_j. Si la cardinalidad del bitset resultante es mayor que la original, se realiza el movimiento. En caso contrario, el intercambio se omite, eliminando el paso 8 del algoritmo. La sección V analiza los efectos que tiene esta búsqueda local eficiente en el algoritmo completo *GRASP*.

IV. *Iterated Greedy*

Iterated greedy (IG) es una metaheurística propuesta por primera vez en 2007 [18], que se ha convertido en una de las más usadas en los últimos años. Las bases de *IG* residen en explorar un espacio de búsqueda alternando intensificación y diversificación iterativamente gracias a sus dos fases, la de destrucción y la de reconstrucción.

En esta propuesta se aplica *IG* sobre la mejor solución encontrada tras ejecutar las dos fases de *GRASP*. El algoritmo consiste en una primera fase de destrucción que resulta en una solución parcial, seguida de una reconstrucción en la que se consigue volver a obtener una solución completa con k elementos en ella. Finalmente, a la solución reconstruida se le aplica un proceso de optimización local. Estos tres pasos se ejecutan iterativamente en caso de que no se haya superado el número máximo de iteraciones sin mejora γ, parámetro que se analiza en la sección V.

Para poder realizar la destrucción se debe establecer el parámetro de entrada β, que determina el porcentaje de elementos de la solución que será eliminado y cuyo valor elegido se determina de forma experimental en la sección V.

Por otro lado, se debe especificar cómo será cada una de las fases del *IG* implementado. Para ello se proponen dos alternativas: aleatoria y voraz. Es por esto que el algoritmo *IG* propuesto tiene 4 combinaciones diferentes variando el método aleatorio y el voraz tanto en el caso de la destrucción como en el de la reconstrucción. La combinación elegida y el motivo se analizan en la sección V.

En el caso del método aleatorio para la destrucción se realiza seleccionando elementos aleatorios de forma iterativa para quitarlos de la solución hasta eliminar $\beta \times k$ elementos. La reconstrucción aleatoria actúa de manera similar, seleccionando elementos iterativamente de forma aleatoria hasta volver a obtener una solución factible.

Para aplicar el procedimiento voraz se vuelve a usar la representación de la solución explicada en la sección II, realizando la operación lógica `AND` para determinar el mejor elemento a añadir en cada iteración en el caso de la reconstrucción. Por otro lado, para destruir la solución se eligen los elementos que menos características tengan en común con el resto de elementos de la solución.

V. Experimentos y resultados

Esta sección tiene dos objetivos principales. Por un lado, determinar qué parámetros son los más adecuados para el algoritmo propuesto. Por otro, es necesario evaluar el rendimiento del algoritmo desarrollado comparándolo con el mejor método encontrado en el estado del arte. Para cumplir los dos objetivos se dividen los experimentos en dos fases: los experimentos preliminares y los finales.

El algoritmo propuesto se ha implementado en Java 11, y todos los experimentos se han realizado en un AMD EPYC 7282 (2.8 GHz) y 8 GB RAM. El conjunto de instancias a la hora de realizar las comparaciones son las usadas en [11], que es el mejor método encontrado en la literatura, para poder llevar a cabo una comparativa justa. Consiste en un total de 238 instancias donde el número de elementos y características va de 32 hasta 300 (ver [11] para más detalles sobre estas instancias).

Todos los experimentos se evalúan según las siguientes métricas: Promedio, la media del valor de la función objetivo; Tiempo (s), el tiempo total de cómputo medido en segundos; Desv. (%), la desviación porcentual media con respecto al mejor resultado del experimento; y # Mejores, el número de soluciones mejores encontradas en el experimento.

La fase de experimentos preliminares está diseñada para ajustar los parámetros de entrada de cada algoritmo. Con el objetivo de no sobreajustar los parámetros, en estos experimentos se considera un subconjunto representativo de 27 del total de 238 instancias.

El primer parámetro estudiado es α del constructivo *GRASP*, donde se han evaluado $\alpha = \{0.25, 0.50, 0.75, \textit{RND}\}$. En concreto, se han ejecutado 1000 iteraciones del algoritmo *GRASP* completo. La Tabla II muestra los resultados obtenidos en este experimento.

α	Promedio	Tiempo (s)	Desv. %	# Mejores
0.25	24.26	**14.66**	5.31 %	18
0.50	22.37	17.26	25.97 %	11
0.75	21.63	17.80	28.94 %	10
RND	**25.07**	15.99	**0.66 %**	**25**

Tabla II: Comparativa de los diferentes valores de α considerados construyendo y mejorando 1000 soluciones independientes.

En este experimento, el mejor valor obtenido es $\alpha = \textit{RND}$ con una desviación del 0.66 % y 25 de las 27 mejores soluciones encontradas. Una vez determinado esto, se analiza el impacto de la búsqueda local eficiente, comparando esta variante con la básica. Los resultados indican que la búsqueda eficiente es 10 veces más rápida, reduciendo su tiempo de ejecución a 1.20 segundos frente a los 15.99 segundos originales, obteniendo los mismos resultados respecto a la calidad.

En cuanto a *Iterated Greedy*, se deben establecer los parámetros β y el número de iteraciones sin mejora. Cabe destacar que la solución de partida de *Iterated Greedy* es siempre la mejor de las 1000 iteraciones GRASP realizadas. A la hora de ajustar el primer parámetro se evalúa $\beta = \{0.1, 0.2, 0.3, 0.4, 0.5\}$, llegando a la conclusión de que el mejor valor se obtiene con $\beta = 0.2$, produciéndose a partir de este valor un estancamiento en la calidad mientras continúa aumentando el tiempo de ejecución. Por otro lado, en cuanto al número de iteraciones sin mejora, se estudian los valores $\gamma = \{5, 10, 15, 20\}$, convergiendo el algoritmo con $\gamma = 10$.

Tal y como se indica al inicio de la sección, el experimento final tiene como objetivo el análisis del rendimiento del algoritmo propuesto frente al mejor resultado encontrado en la literatura. En concreto, el mejor algoritmo del estado del arte se propone en [11], donde los autores presentan el algoritmo *Variable Neighborhood Search (VNS)* reactivo (*Reactive VNS*) descrito en la Sección I. La Tabla III muestra los resultados obtenido por el algoritmo propuesto (*GRASP+IG*) frente a *Reactive VNS*, donde el mejor resultado de cada métrica se resalta en negrita.

Algoritmo	Promedio	Tiempo (s)	Desv. (%)	# Mejores
Reactive VNS	32.59	21.01	2.04	186
GRASP+IG	**33.00**	**0.85**	**0.57**	**225**

Tabla III: Comparativa entre el algoritmo propuesto (*GRASP+IG*) y el estado del arte (*Reactive VNS*) sobre el total de 238 instancias.

Como se puede observar, *GRASP+IG* obtiene mejores resultados en todas las métricas evaluadas. En concreto, obtiene 225 mejores soluciones frente a las 186 obtenidas por *Reactive VNS*. Cabe destacar que, de esas mejores soluciones encontradas, *GRASP+IG* es estrictamente mejor que *Reactive VNS* en 52 de ellas, mientras que *Reactive VNS* solo obtiene 14 soluciones estrictamente mejores. Además, la desviación obtenida del 0.57 % respecto al 2.04 % de *Reactive VNS* indica que, en aquellas soluciones en las que no es capaz de alcanzar el mejor valor, se queda muy cerca del mismo. Por último, y no menos importante, hay que resaltar la relevancia de la representación de la solución descrita en la Sección II, así como el impacto de la búsqueda local eficiente, que llevan a *GRASP+IG* a requerir, en promedio, menos de un segundo por instancia, mientras que *Reactive VNS* necesita más de 20 segundos por cada instancia analizada. Finalmente, se ha ejecutado el test no-paramétrico de Wilcoxon para confirmar que hay diferencias estadísticamente significativas entre los resultados de ambos algoritmos. El p-valor obtenido, menor que 0.001, confirma la hipótesis inicial. Estos resultados confirman la superioridad del algoritmo propuesto para resolver el problema del kMIS.

VI. Conclusiones

El algoritmo propuesto, basado en *Iterated Greedy*, con una construcción inspirada en *Greedy Randomized Adaptive Search Procedure*, es capaz de obtener resultados prometedores tanto en calidad como en tiempo de cómputo. La representación alternativa de la solución propuesta, junto con un diseño inteligente de la búsqueda local, consigue llevar al algoritmo a obtener resultados muy prometedores en tiempos de cómputo despreciables (menos de un segundo por instancia), lo que convierte al algoritmo propuesto en uno de los métodos más competitivos para resolver el kMIS.

Como trabajo futuro se propone la propuesta de funciones objetivo alternativas que permitan enriquecer el espacio de búsqueda. Además, debido al tiempo tan reducido de cómputo requerido por el algoritmo propuesto, sería interesante probar nuevas estrategias de búsqueda, como Tabú Search, que permita ampliar la región del espacio de búsqueda explorado. Estas nuevas estrategias eventualmente llevarán al algoritmo a encontrar mejores soluciones.

Referencias

[1] R. Peeters, "The maximum edge biclique problem is np-complete," *Discrete Applied Mathematics*, vol. 131, no. 3, pp. 651–654, 2003.

[2] A. Pandey, G. Sharma, and N. Jain, "Maximum weighted edge biclique problem on bipartite graphs," in *Conference on Algorithms and Discrete Applied Mathematics*. Springer, 2020, pp. 116–128.

[3] Y. Wang, S. Cai, and M. Yin, "New heuristic approaches for maximum balanced biclique problem," *Information Sciences*, vol. 432, pp. 362–375, 2018.

[4] M. Li, J.-K. Hao, and Q. Wu, "General swap-based multiple neighborhood adaptive search for the maximum balanced biclique problem," *Computers & Operations Research*, vol. 119, p. 104922, 2020.

[5] S. A. Vinterbo, "A Note on the Hardness of the k-Ambiguity Problem," Decision Systems Group/Harvard Medical School, 75 Francis Street, Boston, MA 02115, USA, Tech. Rep. DSG-TR-2002-006, May 2002.

[6] D. Nussbaum, S. Pu, J.-R. Sack, T. Uno, and H. Zarrabi-Zadeh, "Finding maximum edge bicliques in convex bipartite graphs," in *International Computing and Combinatorics Conference*. Springer, 2010, pp. 140–149.

[7] E. T. Bogue, C. C. de Souza, E. C. Xavier, and A. S. Freire, "An integer programming formulation for the maximum k-subset intersection problem," in *Combinatorial Optimization*, P. Fouilhoux, L. E. N. Gouveia, A. R. Mahjoub, and V. T. Paschos, Eds. Cham: Springer International Publishing, 2014, pp. 87–99.

[8] E. C. Xavier, "A note on a maximum k-subset intersection problem," *Information Processing Letters*, vol. 112, no. 12, pp. 471–472, 2012.

[9] B. Ganter and K. Reuter, "Finding all closed sets: A general approach," *Order*, vol. 8, no. 3, pp. 283–290, 1991.

[10] V. Acuña, C. E. Ferreira, A. S. Freire, and E. Moreno, "Solving the maximum edge biclique packing problem on unbalanced bipartite graphs," *Discrete Applied Mathematics*, vol. 164, pp. 2–12, 2014.

[11] F. C. Dias, W. A. Tavares, and J. R. de Freitas Costa, "Reactive VNS algorithm for the maximum k-subset intersection problem," *Journal of Heuristics*, vol. 26, no. 6, pp. 913–941, 2020.

[12] N. Mladenović and P. Hansen, "Variable neighborhood search," *Computers & operations research*, vol. 24, no. 11, pp. 1097–1100, 1997.

[13] T. A. Feo and M. G. Resende, "A probabilistic heuristic for a computationally difficult set covering problem," *Operations Research Letters*, vol. 8, no. 2, pp. 67–71, 1989.

[14] T. A. Feo, M. G. Resende, and S. H. Smith, "A greedy randomized adaptive search procedure for maximum independent set," *Operations Research*, vol. 42, no. 5, pp. 860–878, 1994.

[15] M. G. C. Resende and C. C. Ribeiro, *Greedy Randomized Adaptive Search Procedures: Advances and Extensions*. Cham: Springer International Publishing, 2019, pp. 169–220.

[16] Y. Gao, X. Gao, X. Li, B. Yao, and G. Chen, "An embedded GRASP-VNS based two-layer framework for tour recommendation," *IEEE Transactions on Services Computing*, 2019.

[17] A. Saad, A. Kafafy, O. Abd-El-Raof, and N. El-Hefnawy, "A GRASP-Genetic metaheuristic applied on multi-processor task scheduling systems," in *2018 13th International Conference on Computer Engineering and Systems (ICCES)*. IEEE, 2018, pp. 109–115.

[18] R. Ruiz and T. Stützle, "A simple and effective iterated greedy algorithm for the permutation flowshop scheduling problem," *European journal of operational research*, vol. 177, no. 3, pp. 2033–2049, 2007.

Optimización multiobjetivo basada en el algoritmo ABC para la codificación de proteínas

Belen Gonzalez-Sanchez
Depto. Tecn. Computadores y Comunic.
Universidad de Extremadura
Cáceres, España
belengs@unex.es

Miguel A. Vega-Rodríguez
Depto. Tecn. Computadores y Comunic.
Universidad de Extremadura
Cáceres, España
mavega@unex.es

Sergio Santander-Jiménez
Depto. Tecn. Computadores y Comunic.
Universidad de Extremadura
Cáceres, España
sesaji@unex.es

***Resumen*—La codificación de proteínas para maximizar sus niveles de expresión es un objetivo importante en biología sintética, y añadir múltiples copias del gen que codifica la proteína es una de las técnicas más seguidas para aumentar dichos niveles de expresión. Sin embargo, la codificación de una proteína con múltiples copias lleva asociado un riesgo: inducir la recombinación homóloga entre subsecuencias idénticas que se encuentren cercanas. Este proceso biológico tiene un efecto perjudicial ya que la consecuencia es la pérdida de genes. Por esta razón, al diseñar la codificación de una proteína en varias copias se premia que cada una de las copias o secuencias codificadas sea lo más diferente posible a las demás y también en sí misma, evitando subsecuencias idénticas en una misma secuencia. La codificación de una proteína en múltiples copias creando diferencias entre ellas es posible usando los codones sinónimos que representan a cada aminoácido. No obstante, existe preferencia por el uso de los codones sinónimos con una mayor frecuencia de uso (mejor adaptados a ese organismo). Esta contraposición entre usar los codones sinónimos mejor adaptados y, a la vez, crear diferencias entre secuencias define objetivos opuestos a optimizar durante el proceso de codificación de proteínas. En este trabajo, diseñamos un algoritmo multiobjetivo basado en ABC (*Artificial Bee Colony*) para resolver la codificación de proteínas en múltiples copias. Los resultados obtenidos son competitivos con respecto a los de otros autores.**

***Palabras clave*—Optimización multiobjetivo, algoritmo ABC, codificación de proteínas.**

I. Introducción

Maximizar los niveles de expresión de las proteínas se ha convertido en uno de los objetivos más relevantes de la biología sintética. Con esta finalidad, una de las técnicas más prometedoras es la integración de varias copias codificadas del mismo gen en un organismo. Según la literatura consultada [1], es posible que el nivel de expresión de una proteína aumente proporcionalmente al número de copias integradas, siendo una técnica muy utilizada [2]–[4].

La integración de múltiples copias del gen en un organismo es una tarea compleja y con un alto coste. Por eso, algunos autores sugieren integrar las múltiples copias (secuencias) dentro del organismo próximas entre ellas como método para reducir estos contratiempos [2]–[4]. Sin embargo, esta propuesta presenta una debilidad: cuando secuencias o subsecuencias repetidas están muy próximas aumenta la probabilidad de inducir el proceso genético de recombinación homóloga, que tiene como consecuencia la posible pérdida de secuencias genéticas [5]. Por ejemplo, si tenemos 5 genes concatenados secuencialmente (g_1, g_2, g_3, g_4, g_5) y se produce recombinación homóloga entre el primer y último gen, los genes intermedios (g_2, g_3 y g_4) podrían perderse.

Por este motivo, al codificar una misma proteína con múltiples secuencias (CDS, *CoDing Sequence*) es importante que éstas sean lo más diferentes posible entre ellas y, además, que sean lo más diversas posible en sí mismas, evitando así subsecuencias repetidas a lo largo de una misma CDS. Aunque no hay una longitud exacta ni genérica de las secuencias que inducen recombinación homóloga [6]–[8], todos los estudios coinciden en reducir lo máximo posible la longitud de estas subsecuencias comunes o repetidas. Por esta razón, dos de las funciones objetivo de nuestro problema de optimización multiobjetivo están dedicadas a ello: por un lado, a minimizar la longitud de la subsecuencia más larga que se repite dentro de una CDS o entre un par de CDSs, y por otro lado, a maximizar las diferencias entre pares de CDSs.

Codificar múltiples secuencias que representan a una misma proteína y que sean diferentes entre ellas es factible ya que para codificar los 20 aminoácidos existen 61 codones (conjunto de 3 nucleótidos). Esto significa que cada aminoácido puede ser codificado por uno o varios codones sinónimos. Solamente dos aminoácidos están representados por un único codón y, en el extremo opuesto, otros tienen hasta 6 codones sinónimos. De esta forma, jugando con las distintas combinaciones, podemos obtener diferentes secuencias que representan la misma proteína. Sin embargo, esta pluralidad de codones sinónimos no es tan trivial de combinar, ya que unos codones sinónimos están más adaptados al (tienen una mayor frecuencia de uso en el) organismo que otros. El uso de los codones sinónimos más adaptados tiene una influencia directa sobre el nivel de expresión de la proteína [9] y, por esta razón, es preferible codificar las secuencias usando los codones con mayor frecuencia de uso, es decir, con el mejor CAI (*Codon Adaptation Index*). De esta preferencia se deduce el tercer objetivo a optimizar: maximizar el uso de los codones más adaptados. Esto es completamente contradictorio con los objetivos planteados previamente de crear diferencias entre secuencias. En definitiva, se plantea una optimización multiobjetivo donde mejorar un objetivo implica empeorar otros y viceversa.

En los últimos años, se están realizando diversas investigaciones sobre la optimización del índice de adaptación de los codones (CAI) en codificación de secuencias individuales, teniendo como resultado de ello la aparición de varias aplicaciones web: COOL [10], D-Tailor [11] u OPTIMIZER [12]. Además, otros estudios recientes como [9] demuestran la relación entre el uso de codones más adaptados y genes altamente expresados o [13] que analiza la implicación del sesgo de uso de codones según su nivel de adaptación. No obstante, nuestra propuesta, además de optimizar el uso de codones más adaptados, codifica múltiples secuencias diferentes para una misma proteína. Con esta misma finalidad, encontramos un trabajo publicado recientemente [14] que propone el uso del algoritmo NSGA-II y cuyos resultados tomamos como referencia.

En particular, nuestra propuesta se basa en el algoritmo ABC (*Artificial Bee Colony*) [15], que ha sido utilizado con éxito en otros problemas [16]. En nuestro caso, lo hemos adaptado al contexto multiobjetivo (creando el algoritmo MOABC, *Multi-Objective Artificial Bee Colony*) para optimizar el diseño de múltiples secuencias que codifican una misma proteína evitando inducir recombinación homóloga y, a su vez, usando los codones sinónimos con mejor CAI.

El resto del artículo queda organizado en las siguientes secciones. La Sección II incluye la definición formal del problema multiobjetivo y las funciones objetivo. A continuación, en la Sección III se detalla la implementación de MOABC para abordar el problema planteado. En la sección IV se describen los experimentos realizados, los resultados obtenidos y la comparativa con resultados de otras propuestas. Por último, la Sección V presenta las conclusiones de nuestro trabajo así como posibles trabajos futuros.

II. Definición del problema

En este problema de optimización multiobjetivo, una solución es la codificación de una proteína en I secuencias (CDSs). Como resultado de la codificación, cada secuencia es una cadena de caracteres de longitud L. Dado que trabajamos con secuencias de ARN, cada uno de estos caracteres es un nucleótido (A, C, G, U). El número de secuencias (CDSs) empleadas para codificar la proteína es determinado por el usuario. La Figura 1 muestra un ejemplo de solución para una supuesta proteína formada por 4 aminoácidos y codificada con 3 copias (CDSs).

Cada solución es evaluada por tres funciones objetivo definidas matemáticamente para optimizar dos características biológicas que influyen directamente en el nivel de expresión de la proteína. La primera función es referente a la elección de los codones y el uso de los codones sinónimos más frecuentes (adaptados). Los otras dos funciones corresponden al objetivo de crear diferencias entre todas las CDSs que forman la solución, además de diferencias dentro de cada CDS, evitando así la recombinación homóloga.

II-A. Índice de adaptación de los codones (CAI)

La primera función objetivo está dirigida a optimizar la codificación con los codones sinónimos con mayor frecuencia de uso (más adaptados). Cada codón sinónimo que representa a un aminoácido tiene un frecuencia de uso diferente, siendo más beneficiosos aquellos que tienen una frecuencia de uso más alta.

Para cada CDS i de una solución con N codones, se calcula el índice de adaptación de los codones (CAI) siguiendo la Ecuación 1.

$$CAI(CDS_i) = \sqrt[N]{\prod_{n=1}^{N} W(codon_{i,n})}, \quad (1)$$

donde W es el peso asignado al $codon_{i,n}$ (n-ésimo codón del CDS_i). El peso (en el rango [0,1]) de cada codón sinónimo se ha calculado respecto a las frecuencias de uso resultantes de otro trabajo [14].

Después de calcular el CAI para cada CDS, se obtiene el valor mínimo de entre todos los CAI (todas las CDSs) como indica la Ecuación 2. Este objetivo busca maximizar dicho valor. El hecho de optimizar el mínimo y no otra medida, como por ejemplo la media, evita que valores de CAI bajos queden ocultos entre una buena media general. El objetivo es que todas las CDSs de una solución tengan el valor más alto posible de CAI.

$$mCAI = \min_{1 \le i \le I} \quad CAI(CDS_i). \quad (2)$$

II-B. Distancia de Hamming (HD)

La segunda función objetivo tiene como propósito la optimización de la distancia de Hamming (HD, *Hamming Distance*) entre secuencias. Para cada posible combinación de pares (i, j) de CDSs, se calcula la distancia de Hamming como indica la Ecuación 3. Cuando la posición k de ambas secuencias sean diferentes σ toma el valor de 1 y en caso contrario 0.

$$HD(CDS_i, CDS_j) = \sum_{1 \le k \le L} \sigma(CDS_{i,k}, CDS_{j,k}). \quad (3)$$

Como en el caso anterior, se busca maximizar el mínimo valor de HD encontrado entre todas las CDSs, como muestra la Ecuación 4, previniendo así que valores de HD bajos queden ocultos con el uso, por ejemplo, de la media. Obsérvese que en la Ecuación 4 las distancias de Hamming han sido normalizadas (en el rango [0,1]) con la longitud L.

$$mHD = \min_{1 \le i < j \le I} \quad \frac{HD(CDS_i, CDS_j)}{L}. \quad (4)$$

II-C. Longitud de las cadenas repetidas o comunes (LRCS)

La tercera función objetivo cuantifica la longitud de las subsecuencias idénticas encontradas entre distintas CDSs o dentro de una misma CDS (LRCS, *Length of Repeated or Common Substring*) como indica la Ecuación 5. En la Figura 1 se muestra un ejemplo del cálculo de este objetivo. En dicha

CDS_1	AG**A**	**GCG**	**UC**A	GUA
CDS_2	CGA	GCU	**AGC**	**GUC**
CDS_3	**CGG**	GCA	U**CG**	**G**UU
Aminoácidos	R	A	S	V

Figura 1. Ejemplo de una supuesta proteína de 4 aminoácidos codificada con 3 CDSs. Además ejemplifica el cálculo de la función LRCS. En rojo se muestra la subsecuencia repetida más larga (CGG, en la CDS_3) y en azul la subsecuencia común más larga ($AGCGUC$, entre la CDS_1 y la CDS_2).

solución se destaca la subsecuencia repetida (en rojo) y común (en azul) más larga encontrada. En ese ejemplo, debido a su mayor longitud, el valor que toma $MLRCS$ es la longitud de la subsecuencia común entre la CDS_1 y la CDS_2 (6 nucleótidos).

$$LRCS(CDS_i, CDS_j) = \underset{1 \leq p,q,l \leq L}{length(S_{i,p,l})\ when\ (S_{i,p,l} = S_{j,q,l})}, \quad (5)$$

donde $S_{i,p,l}$ es una subsecuencia de la CDS i, que empieza en la posición p y tienen una longitud de l nucleótidos. Obsérvese que si $p = q$ entonces $i \neq j$.

Como en el caso de las funciones anteriores y con el mismo razonamiento, el objetivo es minimizar el máximo valor de LRCS encontrado en la solución, como indica la Ecuación 6. Nótese que la longitud de las subsecuencias está normalizada (en el rango [0,1]) con la longitud completa L.

$$MLRCS = \underset{1 \leq i \leq j \leq I}{\text{máx}} \frac{LRCS(CDS_i, CDS_j)}{L}. \quad (6)$$

III. Algoritmo multiobjetivo de colonia de abejas artificiales

En este trabajo proponemos una adaptación multiobjetivo del algoritmo de optimización basado en el comportamiento de las abejas (ABC, *Artificial Bee Colony* [15]). El éxito de ABC [16] en muy diversos problemas del mundo real ha motivado su uso y adaptación en este trabajo. ABC define un conjunto de operaciones para alcanzar la mejor solución simulando la búsqueda inteligente de alimento que realizan de forma natural las abejas. Cada abeja de la población (colonia) pertenece a un grupo que lleva a cabo un trabajo específico:

- Obreras: tienen como labor explotar las fuentes de alimento (soluciones) conocidas.
- Observadoras: escogen y explotan las mejores fuentes de alimento (soluciones) en función de la información suministrada por las obreras.
- Exploradoras: tienen como tarea explorar nuevas fuentes de alimento (soluciones), reemplazando las ya agotadas.

En los problemas de optimización multiobjetivo la solución no es única y el resultado de la optimización es un conjunto de soluciones llamado frente de Pareto. Adaptar el algoritmo ABC a nuestro problema con 3 funciones objetivo da lugar al algoritmo multiobjetivo MOABC.

El Algoritmo 1 describe el pseudocódigo de MOABC. Primero, un conjunto de soluciones son inicializadas (línea 2) y forman la población sobre la que se realizan iteraciones hasta alcanzar el número máximo de ciclos (max_cycles). Todas las soluciones son inicializadas de forma aleatoria excepto una de ellas, que es inicializada de forma avariciosa utilizando siempre los codones sinónimos más adaptados y obteniendo la solución con el máximo valor (1) en el objetivo mCAI.

Algoritmo 1 Pseudocódigo del algoritmo MOABC.

Require: $colony\_size$ (número de soluciones), max_cycles (número máximo de generaciones), $limit$ (criterio de abandono) y P_m (probabilidad de mutación)
Ensure: $nondominated\_file$
1: $nondominated\_file \leftarrow \emptyset$
2: $initialize\_colony(colony\_size)$
3: **for** $cycle \leftarrow 1, max\_cycles$ **do**
4: $send\_employed\_bees(colony\_size, P_m)$
5: $rank\_and\_crowding(colony\_size)$
6: $calculate\_selection\_probabilities(colony\_size)$
7: $send\_onlooker\_bees(colony\_size, P_m)$
8: $send\_scout\_bees(limit, cycle)$
9: $rank\_and\_crowding(2 * colony\_size)$
10: $update\_file\_solutions(nondominated\_file)$
11: **end for**

A continuación se inicia un bucle donde en cada iteración se realizan las tareas de las abejas obreras, observadoras y exploradoras. En primer lugar, cada obrera (línea 4) aplica sobre su solución el operador de mutación con probabilidad P_m dando lugar a una solución mutada. Posteriormente, cada observadora (línea 7) escoge como punto de partida a una solución con alta probabilidad (línea 6) de ser buena solución. Para ello, las soluciones previamente son ordenadas (línea 5) por los operadores *rank*, que realiza una ordenación en función de las relaciones de dominancia entre soluciones, y *crowding*, que premia la diversidad de las mismas [17]. A partir de una buena solución, cada observadora aplica el operador de mutación con probabilidad P_m, creando una solución mutada que es añadida a la población. Así, al finalizar las fases de las obreras y observadoras la población duplica su tamaño. Por último, las exploradoras (línea 8) comprueban cuántas veces ha sido mutada cada solución y, en caso de exceder el número máximo establecido ($limit$), la consideran una solución agotada y la reinician generando una nueva solución aleatoria. Para que esta solución pueda competir con el resto, se le aplican m mutaciones antes de ser añadida a la población, siendo m un número de mutaciones proporcional a la iteración en la que se encuentre el algoritmo.

Antes de comenzar una nueva iteración, la población (con doble tamaño del inicial) es ordenada por calidad con los operadores *rank* y *crowding* (línea 9). Las soluciones dentro de la primera mitad ($colony\_size$) formarán la población inicial de la nueva iteración. Además, se actualiza el fichero

$nondominated\_file$ con las posibles soluciones no dominadas (línea 10).

El operador de mutación implementado define 4 tipos de mutaciones aunque solamente una de ellas, elegida de forma aleatoria, es aplicada en cada mutación con el fin de mejorar la calidad de la solución:

1. Cada codón de cada CDS es sustituido por otro de sus sinónimos aleatoriamente con una probabilidad P_m.
2. Cada codón de la CDS con el valor mínimo de CAI es sustituido por otro sinónimo con mayor frecuencia de uso con probabilidad P_m. Si el sinónimo que está codificado en la secuencia ya es el mejor, y por lo tanto, no hay otro sinónimo que mejore el índice, entonces no se produce el cambio en ese codón.
3. Cada codón de una de las CDSs que forman el par con el valor mínimo de HD es sustituido por otro sinónimo con probabilidad P_m.
4. Cada codón que forma la cadena idéntica repetida o común más larga es sustituido por otro sinónimo con probabilidad P_m.

El algoritmo MOABC ha sido implementado en C/C++ para la ejecución de un conjunto de experimentos basados en proteínas reales que se analizan en la siguiente sección y donde se muestran y comparan los resultados obtenidos.

IV. Experimentos y resultados

Para realizar los experimentos hemos utilizado 9 proteínas reales (obtenidas de UniProt, *Universal Protein Resource*), mostradas en la Tabla I teniendo en cuenta la longitud (en aminoácidos, AA) y el número de CDSs. Estas características determinan la complejidad de los experimentos por lo que las proteínas seleccionadas tienen un equilibrio entre ambas y muestran escenarios muy variados. Los resultados obtenidos por nuestro algoritmo MOABC son comparados con el trabajo publicado en [14] basado en el algoritmo NSGA-II. Para realizar una comparativa robusta estadísticamente, se han realizado 31 repeticiones de cada experimento.

Tabla I
Proteínas reales usadas en los experimentos.

Código	Nombre	CDSs	Long. (AA)	CDSs*Long.
B7KHU9	PETG_CYAP7	10	38	380
Q88X33	Y1415_LACPL	9	114	1026
A6L9J9	TRPF_PARD8	8	221	1768
Q89BP2	DAPE_BRADU	7	388	2716
Q91X51	GORS1_MOUSE	6	446	2676
B4TWR7	CAIT_SALSV	5	505	2525
B3LS90	OCA5_YEAS1	4	679	2716
A4Y1B6	FADB_SHEPC	3	716	2148
Q5VZP5	DUS27_HUMAN	2	1158	2316

En primer lugar se ha realizado un estudio paramétrico para encontrar la mejor configuración para nuestro algoritmo, utilizando para ello la proteína B7KHU9 por ser la de menor tiempo de ejecución. El algoritmo NSGA-II se ejecutó con la configuración propuesta por sus autores [14]. Para que la comparativa fuera justa, el algoritmo MOABC se configuró con el mismo tamaño de población y número de generaciones que NSGA-II [14]. Más concretamente, tanto $colony\_size$ como max_cycles se han establecido en 100. Además, MOABC tiene otros dos parámetros, que han sido ajustados según se muestra en la Tabla II, donde se indican los valores probados y el valor que dio los mejores resultados. En conclusión, la probabilidad de mutación (P_m) se ha fijado en 5 %, mientras que el número de intentos de mejora hasta considerar que una solución está agotada ($limit$) se ha establecido en 10.

Tabla II
Valores probados para encontrar la mejor configuración, la cual se destaca en negrita.

	Valores testados						
P_m	1,25 %	2,5 %	**5 %**	10 %	20 %	30 %	40 %
$limit$	2	5	**10**	15			

Para aumentar la robustez de la experimentación, las 3 funciones objetivo han sido normalizadas en el rango [0,1]. La Tabla III muestra los valores (nadir e ideales) utilizados para dicha normalización.

Tabla III
Valores nadir e ideales utilizados para la normalización de los objetivos.

Objetivo	Valor nadir	Valor ideal
mCAI	0	1
mHD	0	0,35
MLRCS	1	0

Con el fin de evaluar y comparar la calidad de las soluciones obtenidas por el algoritmo MOABC frente al trabajo encontrado en la literatura [14], hemos empleado dos métricas de calidad que se usan frecuentemente en optimización multiobjetivo: hipervolumen y cobertura del conjunto (*set coverage*).

IV-A. Hipervolumen

El hipervolumen [18] es una métrica unaria que determina la calidad de un frente de Pareto $\mathcal{A}$ calculando el porcentaje del espacio objetivo, definido por los valores nadir e ideal, que es cubierto por el conjunto de soluciones que forman dicho frente de Pareto. En este trabajo, al tratarse de una optimización de 3 objetivos, el espacio definido es tridimensional. El hipervolumen se calcula según la Ecuación 7.

$$HV(\mathcal{A}, r) = Leb\left(\bigcup_{i=1}^{|\mathcal{A}|} h(a_i, r)\right), \quad (7)$$

siendo $|\mathcal{A}|$ la cardinalidad del conjunto $\mathcal{A}$, $h(a_i, r)$ el volumen que forma cada solución a_i ($\in \mathcal{A}$) con el punto de referencia r y Leb la medida de Lebesgue [19].

El hipervolumen ha sido calculado de la misma forma para ambos algoritmos comparados y en la Tabla IV se muestran los resultados obtenidos. Para cada una de las instancias se presenta el valor mediano junto a la desviación cuartílica y, como se observa, nuestro algoritmo MOABC consigue mejor hipervolumen en 8 de las 9 instancias.

Tabla IV
RESULTADOS DEL HIPERVOLUMEN, MOSTRANDO LA MEDIANA$_{\pm DESVIACION_CUARTILICA}$. EN NEGRITA LOS MEJORES RESULTADOS.

Proteína	MOABC	NSGA-II [14]
B7KHU9	**47,71 %**$_{\pm 0,54\,\%}$	43,72 %$_{\pm 0,0014\,\%}$
Q88X33	**48,71 %**$_{\pm 0,29\,\%}$	46,93 %$_{\pm 0,0010\,\%}$
A6L9J9	**53,77 %**$_{\pm 0,16\,\%}$	52,07 %$_{\pm 0,0015\,\%}$
Q89BP2	**57,42 %**$_{\pm 0,15\,\%}$	55,55 %$_{\pm 0,0025\,\%}$
Q91X51	**59,71 %**$_{\pm 0,19\,\%}$	57,68 %$_{\pm 0,0026\,\%}$
B4TWR7	**57,06 %**$_{\pm 0,15\,\%}$	55,89 %$_{\pm 0,0024\,\%}$
B3LS90	**63,73 %**$_{\pm 0,14\,\%}$	62,43 %$_{\pm 0,0018\,\%}$
A4Y1B6	**60,35 %**$_{\pm 0,26\,\%}$	60,03 %$_{\pm 0,0007\,\%}$
Q5VZP5	68,43 %$_{\pm 0,62\,\%}$	**68,48 %**$_{\pm 0,0020\,\%}$
Media	**57,43 %**	55,91 %

Tabla V
RESULTADOS PARA EL TEST DE NORMALIDAD O TEST DE KOLMOGOROV-SMIRNOV Y EL TEST DE LEVENE.

Proteína	Test Kolmogorov-Smirnov		Test Levene
	MOABC	NSGAII [14]	
B7KHU9	0,180	0,200	0,015
Q88X33	0,187	0,200	0,068
A6L9J9	0,200	0,200	0,400
Q89BP2	0,200	0,200	0,019
Q91X51	0,200	0,200	0,147
B4TWR7	0,200	0,200	0,313
B3LS90	0,053	0,147	0,457
A4Y1B6	0,200	0,200	0,020
Q5VZP5	0,122	0,200	0,149

Para confirmar estos resultados de forma estadística se ha realizado un análisis de significancia estadística usando el test paramétrico de ANOVA, en los casos que sea posible, o el test no paramétrico U de Mann-Whitney. En ambos casos, la hipótesis nula afirma que ambos algoritmos son similares y no se encuentran diferencias significativas entre ellos. El nivel de confianza para realizar estos tests estadísticos se ha establecido en 95 % (p-valor de 0,05).

Para poder aplicar ANOVA las muestras (resultados) de ambos algoritmos tienen que seguir una distribución normal y sus varianzas ser homogéneas. Estas características son evaluadas con el test de Kolmogorov-Smirnov y el test de Levene, respectivamente. La Tabla V muestra los resultados para las 9 instancias y ambos algoritmos. Para las instancias B7KHU9, Q89BP2 y A4Y1B6 no es posible aplicar el test de ANOVA ya que ambos algoritmos no tienen varianzas homogéneas en esos casos. Finalmente, la Tabla VI presenta los resultados del análisis estadístico y 7 de las instancias rechazan la hipótesis nula inicial, por lo tanto, se puede asegurar con un 95 % de confianza que las diferencias de hipervolumen entre ambos algoritmos son estadísticamente significativas. Para las instancias A4Y1B6 y Q5VZP5 no se encontró certeza de que existan diferencias significativas en los resultados. Obsérvese que Q5VZP5 era la única instancia en la que NSGA-II [14] había obtenido mejor hipervolumen.

Tabla VI
RESULTADOS DE ANOVA O DEL TEST U DE MANN-WHITNEY, DEPENDIENDO DE LOS TESTS PREVIOS DE NORMALIDAD Y HOMOCEDASTICIDAD, PARA CONFIRMAR DIFERENCIAS ESTADÍSTICAMENTE SIGNIFICATIVAS ENTRE AMBOS ALGORITMOS.

Proteína	ANOVA	U de Mann-Whitney	Dif. Significativas
B7KHU9	–	0,000	Sí
Q88X33	0,000	–	Sí
A6L9J9	0,000	–	Sí
Q89BP2	–	0,000	Sí
Q91X51	0,000	–	Sí
B4TWR7	0,000	–	Sí
B3LS90	0,000	–	Sí
A4Y1B6	–	0,111	No
Q5VZP5	0,907	–	No

IV-B. Cobertura del conjunto

La cobertura del conjunto es una métrica binaria que evalúa la calidad de los frentes de Pareto confrontando dos de ellos (uno de cada algoritmo). Como indica la Ecuación 8, dados dos frentes de Pareto $\mathcal{A}$ y $\mathcal{B}$, la cobertura del conjunto $SC(\mathcal{A},\mathcal{B})$ calcula qué porcentaje de las soluciones que pertenecen a $\mathcal{B}$ son cubiertas por, al menos, una solución de $\mathcal{A}$. Se dice que una solución b es cubierta por otra solución a si existe dominancia débil ($\succeq$), es decir, a domina a b o al menos es igual.

$$SC(\mathcal{A},\mathcal{B}) = \frac{|\{b \in \mathcal{B}; \exists\ a \in \mathcal{A} : a \succeq b\}|}{|\mathcal{B}|}, \qquad (8)$$

donde $|\mathcal{B}|$ es la cardinalidad del conjunto $\mathcal{B}$.

Esta métrica no es simétrica y por lo tanto $SC(\mathcal{A},\mathcal{B}) \neq 1 - SC(\mathcal{B},\mathcal{A})$. Un valor de $SC(\mathcal{A},\mathcal{B})$ igual a 1 indica que todas las soluciones de $\mathcal{B}$ son cubiertas por, al menos, una solución de $\mathcal{A}$, mientras que un valor de 0 indica que ninguna solución es cubierta.

Para el cálculo de la cobertura del conjunto se han utilizado los frentes de Pareto con valor de hipervolumen mediano de cada instancia y algoritmo. La Tabla VII muestra los resultados obtenidos tanto de la cobertura de MOABC sobre NSGA-II [14] como a la inversa. Podemos concluir que los frentes de Pareto obtenidos por MOABC cubren en mayor porcentaje a las soluciones de los frentes obtenidos por NSGA-II [14]. Solamente en una instancia, B4TWR7, NSGA-II [14] consigue mejor porcentaje. De hecho, la media de todos los resultados muestra una clara ventaja a favor de MOABC (más del 32 %).

V. CONCLUSIONES Y TRABAJOS FUTUROS

En este trabajo presentamos una adaptación del algoritmo ABC a la optimización multiobjetivo (MOABC) para resolver el problema de la codificación de proteínas con múltiples copias. Este problema aborda una tarea fundamental en bioinformática, que es la maximización de los niveles de expresión de las proteínas. El problema de optimización se presenta al realizar dicha codificación en múltiples copias teniendo en cuenta dos aspectos biológicos que están en conflicto. Por una parte, el uso de los codones sinónimos mejor adaptados. Por otra parte, se debe evitar la recombinación homóloga. Por

Tabla VII
RESULTADOS DE LA COBERTURA DEL CONJUNTO, COMPARANDO MOABC Y NSGA-II [14]. EN NEGRITA LOS MEJORES RESULTADOS.

Proteína	SC(MOABC,NSGA-II)	SC(NSGA-II,MOABC)
B7KHU9	**62,00 %**	7,75 %
Q88X33	**54,00 %**	0,62 %
A6L9J9	**36,00 %**	5,43 %
Q89BP2	**53,00 %**	0,64 %
Q91X51	**28,00 %**	6,63 %
B4TWR7	23,00 %	**24,86 %**
B3LS90	**34,00 %**	14,62 %
A4Y1B6	**54,00 %**	4,26 %
Q5VZP5	**28,00 %**	14,87 %
Media	**41,33 %**	8,85 %

este motivo, el uso riguroso de los codones sinónimos mejor adaptados no es ideal ya que todas las copias serían idénticas, produciendo la recombinación homóloga. Con el propósito de encontrar el conjunto de soluciones que codifican a una proteína con secuencias (CDSs) lo más diferentes posibles, pero a su vez, usando los codones sinónimos mejor adaptados, se han optimizado 3 objetivos: mCAI, mHD y MLRCS.

Para evaluar la calidad de nuestra propuesta MOABC, los experimentos se han realizado codificando 9 proteínas reales de distintas longitudes, usando además distintos números de CDSs. De esta forma se han usado 9 instancias reales, que representan escenarios muy variados. Con el propósito de realizar una comparación justa con otro método encontrado en la literatura (NSGA-II [14]), tanto nuestro algoritmo MOABC como NSGA-II [14] han usado los mismos valores en dos parámetros de configuración determinantes, como son el tamaño de la población y el número de ciclos o generaciones (ambos iguales a 100). Además, para aportar robustez estadística a los experimentos se han realizado 31 repeticiones de cada experimento.

Analizando los resultados de los experimentos mediante dos métricas de calidad como son el hipervolumen y la cobertura del conjunto (*set coverage*), encontramos que las soluciones obtenidas por MOABC son mejores que las obtenidas por NSGA-II [14] en casi todas las proteínas. Respecto al hipervolumen, MOABC obtiene mejores valores que NSGA-II [14] en 8 de las 9 instancias. Además para 7 de estas 8 instancias se confirma estadísticamente que las diferencias son significativas entre ambos algoritmos, mientras que para las 2 instancias restantes no se hallan estas evidencias. En cuanto a la cobertura del conjunto, de nuevo MOABC obtiene mejores resultados que NSGA-II [14] en 8 de las 9 instancias, obteniendo una mejora media de más del 32 %. Por tanto, podemos concluir que las soluciones obtenidas por MOABC son claramente de mejor calidad que las obtenidas por NSGA-II [14].

Como trabajo futuro, planteamos optimizar la codificación de proteínas en múltiples copias utilizando otros algoritmos multiobjetivo para así probar si es posible obtener aún mejores resultados en este problema. Además, dados los buenos resultados alcanzados por MOABC, se propone como trabajo futuro la aplicación de dicho algoritmo a otros problemas de optimización multiobjetivo dentro del campo de la bioinformática.

AGRADECIMIENTOS

Esta investigación ha sido parcialmente financiada por el Ministerio de Ciencia, Innovación y Universidades y la Agencia Estatal de Investigación (proyecto PID2019-107299GB-I00/AEI/10.13039/501100011033), por la Junta de Extremadura (ayuda GR18090) y por la Unión Europea (Fondo Europeo de Desarrollo Regional).

REFERENCIAS

[1] A. Vassileva, D. A. Chugh, S. Swaminathan, and N. Khanna, "Expression of hepatitis B surface antigen in the methylotrophic yeast Pichia pastoris using the GAP promoter," *Journal of Biotechnology*, vol. 88, no. 1, pp. 21–35, 2001.

[2] P. Gu, F. Yang, T. Su, Q. Wang, Q. Liang, and Q. Qi, "A rapid and reliable strategy for chromosomal integration of gene(s) with multiple copies," *Scientific Reports*, vol. 5, p. 9684, Apr 2015.

[3] K. E. J. Tyo, P. K. Ajikumar, and G. Stephanopoulos, "Stabilized gene duplication enables long-term selection-free heterologous pathway expression," *Nature Biotechnology*, vol. 27, pp. 760–765, Jul 2009.

[4] C. A. Scorer, J. J. Clare, W. R. McCombie, M. A. Romanos, and K. Sreekrishna, "Rapid selection using G418 of high copy number transformants of Pichia pastoris for high-level foreign gene expression," *Bio/Technology*, vol. 12, pp. 181–184, Feb 1994.

[5] R. Aw and K. M. Polizzi, "Can too many copies spoil the broth?" *Microbial Cell Factories*, vol. 12, no. 1, p. 128, Dec 2013.

[6] P. Shen and H. V. Huang, "Homologous recombination in Escherichia coli: dependence on substrate length and homology," *Genetics*, vol. 112, no. 3, pp. 441–457, Mar 1986.

[7] F. K. Khasanov, D. J. Zvingila, A. A. Zainullin, A. A. Prozorov, and V. I. Bashkirov, "Homologous recombination between plasmid and chromosomal DNA in Bacillus subtilis requires approximately 70 bp of homology," *Molecular and General Genetics*, vol. 234, no. 3, pp. 494–497, Sep 1992.

[8] P. Manivasakam, S. C. Weber, J. McElver, and R. H. Schiestl, "Micro-homology mediated PCR targeting in Saccharomyces cerevisiae," *Nucleic Acids Research*, vol. 23, no. 14, pp. 2799–2800, Jul 1995.

[9] I. Frumkin, M. J. Lajoie, C. J. Gregg, G. Hornung, G. M. Church, and Y. Pilpel, "Codon usage of highly expressed genes affects proteome-wide translation efficiency," *Proceedings of the National Academy of Sciences*, vol. 115, no. 21, pp. E4940–E4949, 2018.

[10] J. X. Chin, B. K.-S. Chung, and D.-Y. Lee, "Codon Optimization OnLine (COOL): a web-based multi-objective optimization platform for synthetic gene design," *Bioinformatics*, vol. 30, no. 15, pp. 2210–2212, 2014.

[11] J. C. Guimaraes, M. Rocha, A. P. Arkin, and G. Cambray, "D-Tailor: automated analysis and design of DNA sequences," *Bioinformatics*, vol. 30, no. 8, pp. 1087–1094, 2014.

[12] P. Puigbò, E. Guzmán, A. Romeu, and S. Garcia-Vallvé, "OPTIMIZER: a web server for optimizing the codon usage of DNA sequences," *Nucleic Acids Research*, vol. 35, no. suppl_2, pp. W126–W131, 2007.

[13] A. Uddin and S. Chakraborty, "Codon usage pattern of genes involved in central nervous system," *Molecular Neurobiology*, vol. 56, p. 1737–1748, 2019.

[14] G. Terai, S. Kamegai, A. Taneda, and K. Asai, "Evolutionary design of multiple genes encoding the same protein," *Bioinformatics*, vol. 33, no. 11, pp. 1613–1620, Jun 2017.

[15] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm," *Journal of Global Optimization*, vol. 39, no. 3, pp. 459–471, Apr 2007.

[16] D. Karaboga, B. Gorkemli, C. Ozturk, and N. Karaboga, "A comprehensive survey: artificial bee colony (ABC) algorithm and applications," *Artificial Intelligence Review*, vol. 42, no. 1, pp. 21–57, Jun 2014.

[17] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, Apr 2002.

[18] Y. Cao, B. J. Smucker, and T. J. Robinson, "On using the hypervolume indicator to compare Pareto fronts: Applications to multi-criteria optimal experimental design," *Journal of Statistical Planning and Inference*, vol. 160, pp. 60–74, 2015.

[19] R. G. Bartle, *The Elements of Integration and Lebesgue Measure*. John Wiley & Sons, Inc., 1995.

Impacto Ambiental de la Experimentación con Metaheurísticas: Un Análisis Coste-Beneficio

Pavel Novoa-Hernández
Escuela de Ciencias Empresariales
Universidad Católica del Norte
Coquimbo, Chile
pavel.novoa@ucn.cl

David A. Pelta
Depto. de Ciencias de la Computación e I.A.
Universidad de Granada
Granada, España
dpelta@ugr.es

***Resumen*—Este trabajo tiene por objetivo analizar el impacto medioambiental (huella de carbono) que produce la ejecución de metaheurísticas. Mediante la simulación de un escenario típico en el contexto de la optimización evolutiva, se busca concienciar sobre la necesidad de tener en cuenta dicho impacto a la hora de diseñar experimentos. Con el objetivo de caracterizar apropiadamente a los algoritmos, hemos propuesto un indicador para medir lo que hemos denominado eficiencia ambiental de la metaheurística, que se basa en la agregación de las mejoras relativas de la calidad de las soluciones obtenidas y la emisiones de CO_2. En las simulaciones realizadas se encontró que a) tanto las características del problema como del algoritmo (metaheurística) pueden aumentar significativamente el consumo de electricidad y las emisiones de carbono; b) el rendimiento superior de algoritmos como CMAES puede estar asociado a un coste significativo de emisiones de CO_2; y c) la ejecución de experimentos en determinados países tiene un coste social mayor que en otros. Se sugieren además, algunas estrategias para reducir el impacto medioambiental al realizar experimentos en este campo.**

***Index Terms*—Metaheurística, huella de carbono, experimentos computacionales.**

I. Introducción

Los Objetivos de Desarrollo Sostenible son un llamado universal a la acción para acabar con la pobreza, proteger el planeta y mejorar la vida y las perspectivas de todos, en todas partes. En 2015, todos los Estados miembros de la ONU adoptaron los 17 Objetivos como parte de la Agenda 2030 para el Desarrollo Sostenible, que establece un plan de 15 años para alcanzar los Objetivos (véase https://sdgs.un.org/).

En el núcleo de dichos objetivos hay un enorme número de temas relacionados con: el transporte inteligente, el despliegue de servicios, la promoción de la salud, las sociedades inteligentes, la gestión de desastres y crisis, etc. En cada uno de estos retos es posible reconocer problemas de decisión y optimización que conducen a modelos y marcos de trabajo (ya sean matemáticos, lingüísticos, computacionales, entre otros) que requieren métodos de solución adecuados.

Aunque algunos de esos problemas no requieren una solución tecnológica, muchos otros pueden abordarse utilizando las herramientas que proporciona la Inteligencia Artificial (IA). El aumento de la potencia de cálculo ha dado lugar a un incremento tanto del tamaño de los problemas que se abordan como del número de experimentos que se pueden realizar.

El aprendizaje automático y la optimización son dos de las principales tareas que subyacen a la resolución de esos problemas, pero, al igual que otros, queremos llamar la atención sobre la huella de carbono que produce el procedimiento de resolución por sí mismo (o el procedimiento realizado para obtener el algoritmo de resolución).

Bastan algunos ejemplos para ilustrar la situación actual. En [1], los autores realizaron un estudio sobre el consumo de ejecutar al algoritmo evolutivo Programación Genética (PG) en diferentes configuraciones de hardware y empleando diferentes configuraciones del tamaño de población (soluciones candidatas del algoritmo). La principal conclusión del estudio fue que si bien las configuraciones más potentes de hardware ejecutan en menor tiempo los experimentos, lo hacen a costa de un consumo mucho más elevado que las configuraciones de menor potencia (ej. dispositivos móviles). Sobre el parámetro relacionado con el tamaño de la población, los autores observaron una relación positiva entre este y el consumo energético. Similar a este estudio, en [2] los autores tomaron como base a PG y propusieron un sistema basado en reglas difusas para predecir el nivel de consumo energético del algoritmo en función del dispositivo donde se ejecuta, algunos parámetros del algoritmo y la dificultad del problema a resolver. Los resultados mostraron que el sistema propuesto predice con buena calidad el consumo energético de PG.

En un conocido informe de finales de 2019 [3], los autores cuantificaron "el costo computacional y ambiental de entrenar modelos de redes neuronales profundas para el procesamiento del lenguaje natural (NLP)". El modelo más sencillo tardó 12 hs. y emitió 26 libras de CO_2, mientras que el más grande tardó el equivalente a 274.120 hs. y emitió 26.155 libras de CO_2.

Más recientemente, en [4] los autores afirman que el entrenamiento del modelo GPT-3 (175B) consumió varios miles de petaflop/s-días de computación mientras que la versión anterior GPT-2 (1.5B) , consumió decenas de petaflop/s-días (note que la "B" en 175 y 1.5 significa miles de millones de parámetros). Concluyeron con dos hechos relevantes: a) la necesidad de ser conscientes del coste y la eficiencia de tales modelos, y b) que mientras el entrenamiento es computacionalmente intensivo, el uso del modelo (cuando está entrenado) es barato.

En [5], el autor resume algunos datos y los esfuerzos actuales por comprender el papel que desempeña la IA en el cambio climático, reconociendo tanto el papel de las técnicas de IA como "solucionadoras" de problemas como, en la otra cara de la moneda, su papel como importantes emisoras de carbono.

Varios autores proponen que la eficiencia se convierta en un criterio de evaluación más común para los trabajos de IA, junto con la precisión y otras medidas relacionadas, y pretenden añadir una declaración clara sobre la huella de carbono de la investigación realizada. Un ejemplo puede verse en [6]

En este contexto, el objetivo de este trabajo es analizar el impacto ambiental que tiene la ejecución de metaheurísticas. Más concretamente, planteamos las siguientes dos preguntas de investigación: 1) ¿cuál es el impacto ambiental que tiene un trabajo "estándar" de experimentación con metaheurísticas? y 2) ¿de qué forma podemos caracterizar dicho impacto? Si bien existen experiencias previas como [1], [2] en el ámbito de las metaheurísticas evolutivas, estas se enfocan principalmente en el consumo energético, y no en las emisiones de CO_2. Por lo que nuestra investigación busca contribuir precisamente caracterizando, en términos de emisiones, la eficiencia ambiental de los algoritmos.

Para responder a estas interrogantes, hemos diseñado un experimento que involucra a dos metaheurísticas populares y varios problemas típicos en el contexto de la optimización evolutiva de un solo objetivo, continua, y sin ruido. Durante la ejecución del experimento hemos medido tanto el rendimiento del algoritmo y sus emisiones asociadas. También hemos propuesto una medida para caracterizar lo que hemos denominado como *eficiencia ambiental* de una metaheurística.

II. Cálculo de la huella de carbono

En términos muy sencillos, la huella de carbono de la implementación de un algoritmo específico puede estimarse en dos pasos [7]. En primer lugar, se calcula una estimación del consumo de energía (en kilovatios, kW) teniendo en cuenta (al menos) el consumo de energía de la CPU, el número de CPUs, el consumo de energía de la GPU y el consumo de energía de la memoria, multiplicado por el tiempo de ejecución del algoritmo. Entonces obtenemos el consumo de kW por hora (kWh) Como es evidente, este paso depende del hardware.

En el segundo paso, se multiplica la cantidad de energía por la intensidad de carbono, que es una medida de los gramos de CO_2 emitidos por kWh. En este paso se tiene en cuenta el origen de la electricidad: si el experimento se realiza en una región geográfica en la que la mayor parte de la energía producida procede de fuentes renovables, su huella de carbono será menor que si se realiza en una región en la que la energía procede de una fuente de carbón.

Este tipo de información está disponible, por ejemplo, en https://www.electricitymap.org donde se puede ver en tiempo real la huella de gases de efecto invernadero (en términos de CO_2 equivalente) del consumo de electricidad.

Otro punto importante es el llamado valor de la Eficiencia del Uso de la Energía (PUE), que determina la eficiencia de un centro de datos en la utilización de su energía. Por ejemplo, un PUE de 2.0 significa que por cada 1 kWh de electricidad que llega al servidor, el centro de datos necesita 2 kWh para tener en cuenta los residuos y otros servicios como la refrigeración.

Hoy en día existen varias herramientas, que incluyen bases de datos de especificaciones de hardware, para medir el impacto del carbono de los experimentos computacionales.

Como indican en [8] las emisiones producidas en el entrenamiento de un modelo de red neuronal están relacionadas con la ubicación del servidor de entrenamiento y la red de energía que utiliza, la duración del procedimiento de entrenamiento y el hardware en el que éste se realiza. Los autores proveen una calculadora de emisiones para estimar el uso de energía (disponible en https://mlCO2.github.io/impact/) y el impacto medioambiental concomitante, del entrenamiento de modelos ML.

Una calculadora similar está disponible en https://green-algorithms.org/. La calculadora "se integra fácilmente en los procesos de cálculo, ya que requiere un mínimo de información y no interfiere con el código existente, a la vez que tiene en cuenta una amplia gama de CPUs, GPUs, computación en la nube, servidores locales y ordenadores de sobremesa".

Mientras que las calculadoras anteriores proporcionan estimaciones "gruesas" de la huella de carbono, existen otras técnicas que proporcionan estimaciones más precisas.

Un ejemplo es https://codecarbon.io/. Este paquete de Python permite a los desarrolladores hacer un seguimiento de las emisiones de dióxido de carbono (CO_2) a través de experimentos de aprendizaje automático u otros programas.

Otro es el *experiment-impact-tracker* [6] que es el que se utiliza en este trabajo y se describe a continuación.

III. Diseño de los experimentos

Para medir el impacto medioambiental de la ejecución de experimentos computacionales con metaheurísticas, tomaremos como ejemplos problemas de optimización acotados y mono-objetivo (SOBCO, por sus siglas en inglés) [9]. Se asume que los algoritmos optimizan funciones de caja negra sobre dominios continuos. Para la implementación de los algoritmos y problemas se emplea el *framework* DEAP [10].

Para controlar el consumo de energía, las emisiones de carbono y el coste social de los experimentos, nos basamos en el framework *experiment-impact-tracker* [6]. Este framework de código abierto fue desarrollado en Python y está disponible como repositorio de Github [1]. Originalmente diseñado para el seguimiento de la experimentación de aprendizaje automático, puede ser fácilmente adaptado al contexto de las metaheurísticas. Desde una perspectiva algorítmica, la Fig. 1 muestra cómo utilizamos el *experiment-impact-tracker* en los experimentos. Inicialmente se lanza el rastreador de impacto antes de ejecutar 30 ejecuciones independientes (semillas aleatorias diferentes). Una ejecución se concibe como una simulación donde una instancia de una metaheurística resuelve una instancia de un

[1] https://github.com/Breakend/experiment-impact-tracker

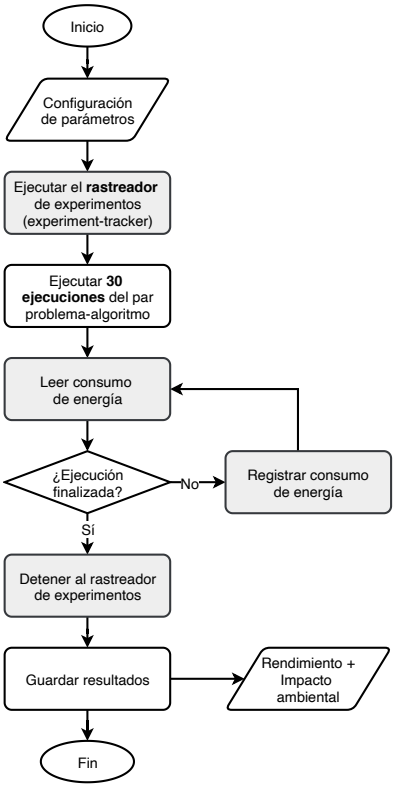


Figura 1. Diagrama de los pasos seguidos para medir el impacto ambiental.

problema. Durante este proceso, el rastreador lee el consumo de energía de varias fuentes (ej. CPU, GPU y DRAM). Además, estos datos se registran en una carpeta específica si las 30 ejecuciones no han terminado. En caso contrario, detenemos el rastreador explícitamente y guardamos la información sobre el impacto y el rendimiento del algoritmo en un archivo. Esta última operación finaliza el programa.

En cada simulación se almacena la siguiente información: el rendimiento final de la metaheurística (media de la mejor solución a partir de 30 ejecuciones) y el impacto de estas ejecuciones en términos de energía consumida (kWh), emisiones de carbono (kgCO_2) y coste social para varios países (dólares por tonelada de CO_2). Los países considerados fueron: China, Estados Unidos, India, Reino Unido, España y Alemania, que son, por ese orden, los seis primeros países donde más investigación se reporta sobre metaheurísticas [2]. También hemos incluido a Chile, por ser el lugar donde se realizaron los experimentos presentados en este trabajo.

Los experimentos se ejecutaron de forma secuencial en un ordenador iMac, con un procesador Intel Core i5 (3 GHz) y 40 GB de RAM. Específicamente, no se realizaron cálculos en la GPU. A continuación describimos el escenario que nos servirá de ejemplo para analizar el impacto ambiental de la experimentación con metaheurísticas y cómo evaluarlo a partir de una propuesta de medida basada en el concepto de eficiencia ambiental.

[2]Según Scopus (http://www.scopus.com) utilizando como consulta de búsqueda: TITLE-ABS-KEY (metaheuristic OR "evolutionary algorithm" OR "evolutionary computation").

III-A. Configuración de los experimentos

El objetivo del escenario de experimentación es comparar el rendimiento de dos metaheurísticas muy populares en el ámbito de la optimización evolutiva: Differential Evolution (DE) [11], y Covariance Matrix Adaptation Evolution Strategy (CMAES) [12]. Hemos considerado solo una configuración de cada algoritmo compuesta por una población (cantidad de soluciones candidatas) igual a $20 \times D$ (donde D es la dimensión del problema). Se consideraron las siguientes 8 funciones (problemas) populares en el ámbito de la computación evolutiva[3]: *Sphere*, *Cigar*, *Rastrigin*, *Ackley*, *Griewank*, *Rosenbrock*, *Bohachevsky* y *Schafer*; para valores de dimensionalidad $D = \{10, 50, 90, 130, 170\}$.

A partir de esta configuración podemos fácilmente hipotetizar que la mayor sobrecarga se producirá en los experimentos que implican instancias de problemas con dimensión más alta ($D = 170$) y funciones objetivo más complejas (por ej. *Bohachevsky* o *Schafer*). Esto se debe a que, por un lado, la dimensión del espacio de búsqueda tiene un efecto multiplicativo en el número de operaciones dentro de los sumatorios incluidos en las funciones objetivo, y por otro lado, las funciones con definiciones complejas involucran más operaciones matemáticas. Como consecuencia de esto último, las unidades de procesamiento implicadas en la ejecución del experimento (por ej. CPU) tendrían que realizar más operaciones aritméticas. En cuanto a las metaheurísticas, esperamos un mayor gasto computacional en los experimentos con CMAES, ya que esta depende de la adaptación de la matriz de covarianza en cada iteración, lo cual se traduce en más operaciones aritméticas.

III-B. Propuesta para el cálculo de eficiencia ambiental

Con el propósito de caracterizar ambientalmente a las metaheurísticas, en esta sección proponemos una medida que involucra tanto la calidad de las soluciones obtenidas como el gasto asociado en obtenerlas. Hemos denominado a dicha medida como *Eficiencia ambiental* y se basa fundamentalmente en un análisis básico de tipo coste-beneficio. De esta forma, concebimos la eficiencia ambiental como la suma de las mejoras relativas del algoritmo en cuanto a la calidad de las soluciones obtenidas, y las emisiones de CO_2:

$$E_{alg,prob} = M_{calidad} + M_{emisiones} \quad (1)$$

donde $M_{calidad}$ y $M_{emisiones}$ son las mejoras relativas que el algoritmo alg alcanza en el problema $prob$ con respecto a la *mejor solución conocida* hasta el momento, en términos de calidad y emisiones de CO_2, respectivamente. Conceptualmente, esta *mejor solución conocida* es una solución de referencia que el investigador puede emplear como base para evaluar una nueva metaheurística. Asumiremos que esta solución de referencia viene representada por $\overline{x}$ y tiene asociados una

[3]https://deap.readthedocs.io/en/master/api/benchmarks.html

calidad $f_{prob}(\overline{x})$ y unas emisiones de CO_2, $\overline{e} = CO2_{prob}(\overline{x})$. En particular, f_{prob} es la función objetivo del problema $prob$ y $CO2_{prob}(x)$ es la función que calcula las emisiones producidas para obtener la solución x.

Para calcular las mejoras relativas (tanto de calidad como de emisiones) de una solución $\widehat{x}$ respecto a la solución de referencia $\overline{x}$ haremos:

$$M_{calidad} = 100\left(1 - \frac{f_{prob}(\widehat{x})}{f_{prob}(\overline{x})}\right) \quad (2)$$

$$M_{emisiones} = 100\left(1 - \frac{CO2_{prob}(\widehat{x})}{CO2_{prob}(\overline{x})}\right) \quad (3)$$

En la Eq. 2 se asume que f_{prob} es la función objetivo a minimizar del problema de optimización en cuestión. En caso de que el problema fuese de maximización, se podría simplemente multiplicar a los f_{prob} por -1. Otro problema que puede surgir con las expresiones anteriores es que la función objetivo puedo tomar valores iguales a 0. En este caso, existen varias opciones. Una de ellas, que hemos aplicado en nuestros experimentos, sería desplazar al rango de valores que puede tomar la función objetivo de manera que el mínimo sea un valor positivo mayor a 0. Por ejemplo, igual a 1.

Es importante notar que las expresiones (2, 3) admiten valores en el rango $(-\infty, \infty)$. Para el caso de la expresión (2), cuando la calidad de la solución de referencia coincida con el de la solución global óptima del problema, entonces el límite de superior de este rango sería 100, en lugar de $+\infty$. Independientemente de lo anterior, los signos de los valores que pueden tomar estos indicadores se interpretan como sigue: un valor positivo indica que la mejora es positiva (el algoritmo mejora a los resultados de la mejor solución conocida), mientras que uno negativo, no (el algoritmo empeora). Por ejemplo, supongamos que la mejor solución conocida para cierto problema p tiene una calidad $f_{prob}(\overline{x}) = 10$ y unas emisiones asociadas de $CO2_{prob}(\overline{x}) = 1$ (kgCO2); y que la mejor solución propuesta por un algoritmo alg tiene una calidad $f_{prob}(\widehat{x}) = 5$ y unas emisiones asociadas de $CO2_{prob}(\widehat{x}) = 3$. Entonces, de acuerdo con la expresión (2) el algoritmo tiene los siguientes valores de mejora:

$$M_{calidad} = 100\left(1 - \frac{5}{10}\right) = 50\,\% \quad (4)$$

$$M_{emisiones} = 100\left(1 - \frac{3}{1}\right) = -200\,\% \quad (5)$$

por lo que se puede concluir que mientras el algoritmo contribuye a mejorar la calidad de la mejor solución conocida en un 50 %, esta contribución fue a costa de aumentar la cantidad de emisiones de CO_2 en un 200 %. Teniendo en cuenta estos dos valores y la expresión (1), la eficiencia de nuestro algoritmo sería igual a $50 + (-200) = -150\,\%$. Dado que el valor obtenido es negativo, puede concluirse que el algoritmo no es eficiente ambientalmente, pues su mejora en la calidad no compensa el costo ambiental de obtener dicha mejora. En otras palabras, su contribución trae más inconvenientes que beneficios.

IV. Resultados

En esta sección presentamos y discutimos los resultados obtenidos. Por cada metaheurística comparamos el nivel de las emisiones de CO_2 que se generaron como producto de sus ejecuciones en cada problema. Además, analizaremos la eficiencia ambiental de la metaheurística con mejor rendimiento (CMAES) tomando como referencia a la de peor rendimiento (DE). Finalmente, resumiremos el costo social por países de todos los experimentos realizados.

IV-A. Emisiones de los experimentos

En la Fig. 2 se resumen las emisiones totales de los experimentos realizados por cada metaheurística y problema. Como se aprecia, nuestras suposiciones se confirman. Por un lado, es apreciable el incremento en emisiones para los experimentos en problemas de mayor dimensionalidad y funciones de definición compleja; y por otro, el alto nivel de emisiones asociado a la metaheurística CMAES comparado con DE. En este último caso, a excepción de los problemas de baja dimensionalidad ($D = 10$), las emisiones de CMAES duplican a las de DE.

Aunque no lo incluimos aquí, el rendimiento de CMAES, en términos de la calidad promedio de las soluciones obtenidas, fue muy superior en comparación con DE en la mayoría de los problemas. Por lo que resulta interesante conocer en qué medida esta superioridad de CMAES compensa su gasto en emisiones.

IV-B. Efectividad de CMAES

En la Fig. 3 se muestran las mejoras en términos de calidad y emisiones que se obtuvieron con la metaheurística CMAES. En cada problema hemos asumido que la *mejor solución conocida* es la peor solución obtenida por DE o CMAES. Por supuesto, el experimentador podría apoyarse en otra solución de referencia. En nuestro caso hemos hecho esta selección con propósitos puramente ilustrativos.

Como se aprecia en la Fig. 3 (a), CMAES mejora la calidad de las soluciones obtenidas por DE en casi todos los problemas. Con excepción de *Rastrigin D = 10*, en el resto de los problemas las mejoras en la calidad de las soluciones son muy cercanas al 100 %. Esto aparece ilustrado en la Fig. 3 (a) por las barras a la derecha del valor 0 %. Contrario a este patrón, también podemos observar en la propia Fig. 3 (a) que CMAES empeora las emisiones asociadas a la mejor solución conocida. Nótese que con excepción de algunos casos, como los problemas con dimensión igual a 10, en la mayoría se obtienen valores negativos (barras ubicadas a la izquierda del valor 0 %).

Resumiendo la información de las mejoras, el gráfico de barras de la Fig. 3 (b) muestra que CMAES es eficiente ambientalmente en una menor cantidad de problemas. Esto se aprecia en las barras que están a la derecha del valor 0 %.

IV-C. Costo social de los experimentos

Finalmente, resulta de interés conocer qué costo social tendrían los experimentos realizados en este ejemplo ilustrativo. Este costo total es interpretado como el daño marginal [13].

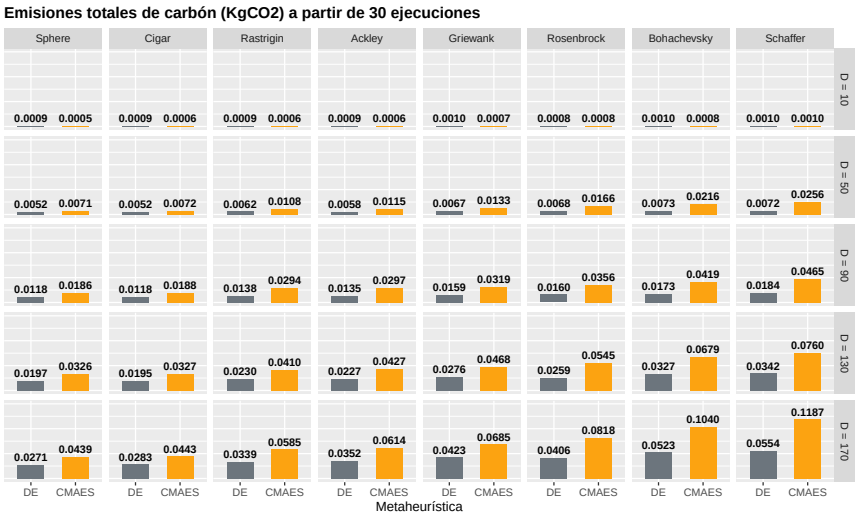


Figura 2. Emisiones de CO_2 estimadas para las metaheurísticas DE y CMAES por cada problema.

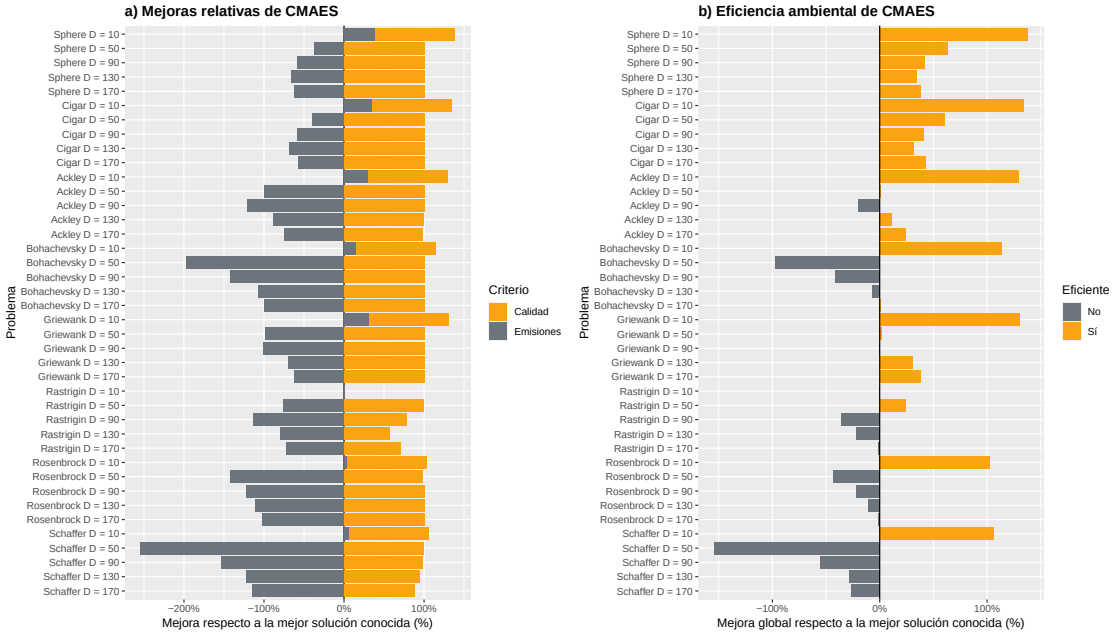


Figura 3. Mejoras relativas y eficiencia ambiental de CMAES en los problemas considerados.

En tal sentido, la Fig. 4 muestra esta información por cada país considerado. Se puede apreciar que, aunque marginales en términos absolutos, los valores de costo social de los experimentos varían sustancialmente entre países. Por ejemplo, en el caso de la India (IND) este valor es particularmente alto comparado con países como Alemania (DEU), Reino Unido (GBR) o Chile (CHL). En estos últimos, de acuerdo con [13], sus valores negativos indican que la realización de estos experimentos en dichos países no causa daños, sino beneficios. Esto se debe a que sus correspondientes temperaturas están "por debajo del óptimo económico" [13].

V. Conclusiones

Toda actividad humana tiene un impacto en el medio ambiente. Aunque el análisis de este impacto ha sido objeto de seguimiento en diversos contextos, aún queda mucho por hacer. En este trabajo contribuimos a cerrar esta brecha analizando el impacto ambiental de la experimentación con metaheurísticas. A partir de las simulaciones realizadas concluímos que:

1. ciertas características del problema a optimizar, (ej. dimensión, función objetivo) son determinantes en el aumento del consumo energético, y por tanto de las emisiones de CO_2, de los experimentos;

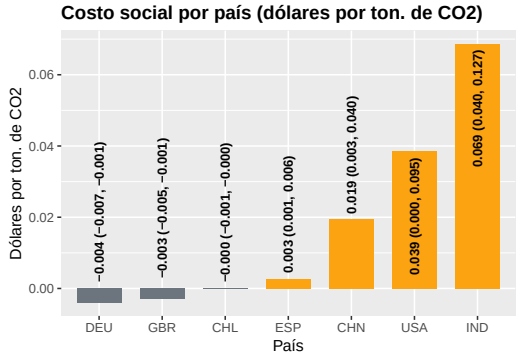


Figura 4. Costo social de los experimentos por países. Las barras del gráfico corresponden a las medianas estimadas (50 %), mientras que los números dentro de los paréntesis a los intervalos de confianza del 66 %.

2. algunas metaheurísticas históricamente eficaces, como CMAES, están asociadas a un mayor consumo de energía y emisiones de CO_2 en comparación con otras de menor complejidad computacional como DE. Esto tiene importantes implicaciones prácticas en escenarios reales, ya que junto al rendimiento de la metaheurística, los profesionales tendrán que considerar otro objetivo aparentemente conflictivo: su impacto medioambiental;
3. de ser posible, sería crucial elegir adecuadamente la ubicación donde se ejecutarán los experimentos, especialmente para los experimentos a gran escala.
4. aunque los bajos valores reportados por nuestras simulaciones pueden sugerir que la experimentación con metaheurísticas no impacta significativamente en el medio ambiente, hay que tener en cuenta que los escenarios considerados representan sólo una pequeña fracción de los actualmente estudiados en este campo, que implican escenarios experimentales mucho más complejos. La optimización robusta en el tiempo [14] es un ejemplo en este caso.

Además de estas conclusiones, sugerimos algunas acciones para reducir el impacto negativo de la experimentación con metaheurísticas. En primer lugar, antes de diseñar experimentos computacionales, hay que tratar de entender lo más posible la teoría que explica el comportamiento de la metaheurística. De este modo, se puede estimar el grado de eficacia de la metaheurística y así ahorrar ejecuciones innecesarias, lo que se traduce en un menor consumo de energía. En este contexto, hay que tener en cuenta también las implicaciones de los teoremas *No Free Lunch* [15]: no hay ninguna metaheurística ganadora cuando se consideran todos los problemas posibles.

En segundo lugar, se deben plantear diseños experimentales que impliquen menos combinaciones de factores a probar que los diseños factoriales completos. La literatura incluye importantes alternativas a considerar [16]. Un aspecto clave aquí más allá de, por ejemplo, los parámetros de la metaheurística, es el punto de referencia considerado. Existe una tendencia a aumentar el tamaño de los *benchmarks*, y muchas veces, esto no está debidamente justificado.

Por último, tal y como se sugiere en [6], es recomendable incluir en los reportes de resultados, una *declaración de carbono* que resuma cómo impactó la experimentación en el ambiente. Esto ayudará a concienciar, al menos en nuestro campo de investigación, sobre el uso racional de los recursos informáticos.

Declaración de control de emisiones

La experimentación desarrollada en este estudio aportó 2.043 kg de CO_{2eq} a la atmósfera y utilizó 0.805 kWh de electricidad, teniendo un coste social del carbono específico de CHL de $-0{,}0003(-0{,}0005, -2{,}3716e-05)$. Se asume que el multiplicador PUE es $1{,}56$.

Agradecimientos

D. Pelta agradece el apoyo del proyecto (TIN2017-86647-P) del Ministerio de de Economía, Industria y Competitividad, España (incluyendo fondos FEDER).

Referencias

[1] F. F. de Vega, F. Chávez, J. Díaz, J. A. García, P. A. Castillo, J. J. Merelo, and C. Cotta, "A cross-platform assessment of energy consumption in evolutionary algorithms," in *Parallel Problem Solving from Nature – PPSN XIV*. Springer International Publishing, 2016, pp. 548–557.

[2] J. Álvarez, F. Chávez, P. Castillo, J. García, F. Rodriguez, and V. de, "A fuzzy rule-based system to predict energy consumption of genetic programming algorithms," *Computer Science and Information Systems*, vol. 15, no. 3, pp. 635–654, 2018.

[3] E. Strubell, A. Ganesh, and A. McCallum, "Energy and policy considerations for deep learning in NLP," 2019.

[4] T. B. Brown and et. al, "Language models are few-shot learners," 2020.

[5] P. Dhar, "The carbon impact of artificial intelligence," *Nature Machine Intelligence*, vol. 2, no. 8, pp. 423–425, aug 2020.

[6] P. Henderson, J. Hu, J. Romoff, E. Brunskill, D. Jurafsky, and J. Pineau, "Towards the systematic reporting of the energy and carbon footprints of machine learning," 2020. [Online]. Available: https://arxiv.org/abs/2002.05651

[7] S. Bergman, "How can I calculate co2eq emissions for my Azure VM?" 2021. [Online]. Available: https://bit.ly/35ypZqx

[8] A. Lacoste, A. Luccioni, V. Schmidt, and T. Dandres, "Quantifying the carbon emissions of machine learning," CoRR, vol. abs/1910.09700,2019. [Online]. Available: http://arxiv.org/abs/1910.097002019.

[9] A. Wagdy, A. A. Hadi, A. K. Mohamed, P. Agrawal, A. Kumar, and P. N. Suganthan, "Problem definitions and evaluation criteria for the CEC 2021 Special Session and Competition on single objective bound constrained numerical optimization," Nanyang Technological University, Technical Report, 2020.

[10] F.-A. Fortin, F.-M. D. Rainville, M.-A. Gardner, M. Parizeau, and C. Gagné, "DEAP: Evolutionary algorithms made easy," *Journal of Machine Learning Research*, vol. 13, no. 70, pp. 2171–2175, 2012.

[11] R. Storn and K. Price, "Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.

[12] N. Hansen, "The CMA evolution strategy: A comparing review," in *Towards a New Evolutionary Computation: Advances in the Estimation of Distribution Algorithms*, J. A. Lozano, P. Larrañaga, I. Inza, and E. Bengoetxea, Eds. Springer Berlin Heidelberg, 2006, pp. 75–102.

[13] K. Ricke, L. Drouet, K. Caldeira, and M. Tavoni, "Country-level social cost of carbon," *Nature Climate Change*, vol. 8, no. 10, pp. 895–900, 2018.

[14] P. Novoa-Hernandez, D. A. Pelta, and C. C. Corona, "Approximation models in robust optimization over time - an experimental study," in *IEEE Congress on Evolutionary Computation (CEC)*, 2018, pp. 1–6.

[15] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 67–82, 1997.

[16] J. Antoy, *Design of Experiments for Engineers and Scientists: Second Edition*. Elsevier, 2014.

Un enfoque multi-objetivo para el problema de la máxima diversidad

Pedro Casas-Martínez
Dept. Computer Science
Universidad Rey Juan Carlos
Móstoles, España
p.casas.2017@alumnos.urjc.es
0000-0002-7859-2988

Alejandra Casado-Ceballos
Dept. Computer Science
Universidad Rey Juan Carlos
Móstoles, España
a.casadoce.2017@alumnos.urjc.es
0000-0003-3417-6859

Jesús Sánchez-Oro
Dept. Computer Science
Universidad Rey Juan Carlos
Móstoles, España
jesus.sanchezoro@urjc.es
0000-0003-1702-4941

Eduardo G. Pardo
Dept. Computer Science
Universidad Rey Juan Carlos
Móstoles, España
eduardo.pardo@urjc.es
0000-0002-6247-5269

***Resumen*—El problema de la máxima diversidad ha sido ampliamente estudiado desde diferentes perspectivas, haciendo uso de técnicas exactas y heurísticas. En este artículo, nos enfrentamos a una variante multi-objetivo del problema, la cual intenta optimizar simultáneamente las cinco métricas de diversidad cuyo uso está más extendido en la literatura. Para lograrlo, proponemos una novedosa adaptación del ya conocido *Greedy Randomized Adaptive Search Procedure*, que ha sido usado tradicionalmente para optimización mono-objetivo. Con este nuevo enfoque, un conjunto de soluciones eficientes es generado siguiendo dos estrategias constructivas diferentes, analizando el rendimiento de ambas. La fase de mejora del algoritmo propuesto consiste en un procedimiento de búsqueda local basado en una estructura de intercambios siguiendo un enfoque *first improvement*. Además, se propone una exploración inteligente de la vecindad que permita limitar el espacio de búsqueda. Los experimentos computacionales muestran la calidad del algoritmo propuesto, comparado con los resultados obtenidos con el mejor método previo de la literatura. Las diferencias encontradas han sido respaldadas por pruebas estadísticas no paramétricas.**

***Index Terms*—diversidad, multi-objetivo, GRASP, búsqueda local**

I. Introducción

La comunidad científica lleva estudiando la familia de problemas sobre la máxima diversidad desde 1988 [9], mientras que los primeros modelos de programación entera fueron introducidos en Dhir et al. [2] y Kuo et al. [10]. El concepto de la diversidad ha evolucionado con el paso del tiempo, y muchas propuestas nuevas en la literatura han mejorado su definición [17] construyendo así una familia de problemas de optimización. El alto interés en este tipo de familia de problemas se debe principalmente a sus aplicaciones prácticas en una gran variedad de áreas: análisis de redes sociales, problemas de localización de instalaciones, generación de un conjunto de estudiantes heterogéneos en clase, etc. Ha sido demostrado que todos estos problemas son problemas de optimización combinatoria $\mathcal{NP}$-difíciles [16]. La idea principal en la que se basan los problemas de optimización de diversidad consiste en seleccionar un subconjunto de elementos de un conjunto dado, con el objetivo de maximizar una cierta medida de distancia. La principal diferencia entre la mayoría de enfoques propuestos es la forma de calcular la diversidad de los elementos seleccionados.

Esta investigación está financiada por el Ministerio de Ciencia, Innovación y Universidades con el proyecto con referencia PGC2018-095322-B-C22, y por la Comunidad de Madrid y los Fondos Estructurales de la Unión Europea con los proyectos con referencias S2018/TCS-4566 y Y2018/EMT-5062.

Antes de definir las métricas estudiadas en este artículo, vamos a explicar algunas definiciones preliminares comunes en la familia de problemas de optimización de la diversidad. Dado un conjunto N de elementos, el objetivo es encontrar un subconjunto S de p elementos, siendo $p < |N|$, de tal forma que la diversidad de los elementos incluidos en S sea maximizada. Para evaluar la diversidad se debe definir una métrica de distancia d_{ij}, con $i, j \in N$, entre dos elementos. Para preservar la generalidad, no consideramos ninguna métrica de distancia específica, sino que cualquier métrica puede ser seleccionada: distancia euclídea, distancia del coseno, índice de similitud de jaccard, entre otras [20, 19]. Tras esto, el problema de la máxima diversidad busca encontrar una solución $S^\star$ que maximiza una cierta métrica de diversidad. Más formalmente,

$$S^\star \leftarrow \arg\max_{S \in \mathcal{SS}} d(S) \tag{1}$$

donde $\mathcal{SS}$ representa el espacio de búsqueda, i.e., el conjunto de todas las posibles combinaciones de p elementos distintos de N.

Los problemas de maximización de la diversidad han sido tradicionalmente abordados desde una perspectiva mono-objetivo, enfocándose en encontrar un subconjunto de los elementos más diversos con respecto a una sola métrica de diversidad. Sin embargo, es interesante analizar los resultados obtenidos al modelar la diversidad como un problema de optimización multi-objetivo. Además, en el contexto de aplicaciones en la vida real, es difícil seleccionar sólo una métrica de diversidad que satisfaga todos los requerimientos del problema. Por tanto, es deseable la capacidad de optimizar simultáneamente más de una métrica. En ese caso, estamos lidiando con un problema de optimización multi-objetivo (MOP, del inglés *Multi-objective Optimization Problem*), en el cual varias funciones objetivo deben ser optimizadas al mismo tiempo. La principal característica al lidiar con un MOP es que la funciones objetivo consideradas están en conflicto, i.e., optimizar una de ellas suele deteriorar otras. Debido a esto,

no existe una solución única que optimiza todos los objetivos al mismo tiempo. Por tanto, el resultado de un MOP no es una única solución sino un conjunto de soluciones eficientes.

El conjunto de soluciones eficientes es construido siguiendo el principio de optimalidad de Pareto, que presenta un balance para una solución en la cual una función objetivo no puede ser mejorada sin deteriorar otra. Por tanto, la comparación de dos soluciones es dada por el concepto de dominancia. Específicamente, dadas dos soluciones S_1 y S_2, podemos afirmar que S_1 domina a S_2 si y solo si la evaluación de S_1, considerando cada función objetivo, es mejor o igual que el valor de la función objetivo correspondiente en S_2. De no ser así, se dice que S_1 y S_2 son recíprocamente no dominadas. Según esta definición, una solución S_1 es considerada eficiente si no existe ninguna otra solución S_2 en el espacio de soluciones tal que S_2 domine a S_1. El conjunto que contiene las mejores soluciones no dominadas del espacio de búsqueda se denomina Frente de Pareto.

Este trabajo está centrado en proponer un algoritmo que sea capaz de optimizar más de una medida de diversidad al mismo tiempo con el objetivo de encontrar diferencias y similitudes entre las medidas de diversidad más extendidas. Este enfoque fue presentado originalmente en Vera et al. [22], donde los autores estudiaron cinco medidas de diversidad diferentes simultáneamente. Este problema fue llamado *Multi-Objective Maximum Diversity Problem* (MOMDP). Concretamente, los autores en Vera et al. [22] propusieron un algoritmo evolutivo multi-objetivo para resolver el MOMDP.

En particular, se propone una adaptación del ya conocido *Greedy Randomized Adaptive Search Procedure* (GRASP) para resolver el MOMDP. El resto del trabajo está organizado de la siguiente forma: la Sección II describe las métricas de diversidad consideradas en este trabajo, la Sección III detalla la propuesta algorítmica, la Sección IV presenta los experimentos computacionales llevados a cabo para analizar el desempeño del algoritmo propuesto y, finalmente, la Sección V detalla las conclusiones más relevantes de esta investigación.

II. Métricas de diversidad

En esta sección se definen las medidas de diversidad estudiadas en este trabajo, las cuales han sido consideradas juntas previamente en [22]. En dicho estudio, los autores muestran como los objetivos propuestos están en conflicto, es decir, no es posible mejorar en uno de ellos sin deteriorar la calidad de los otros.

La primera medida de diversidad es *Max-Sum Diversity (MSD)*. En este caso, la diversidad de un subconjunto de elementos es evaluada como la suma de las distancias entre cada par de elementos seleccionados [15], tal como está definido en la Ecuación 2.

$$MSD(S) = \sum_{\substack{i,j \in S \\ i<j}} d_{ij} \tag{2}$$

El objetivo es, por tanto, encontrar una solución $S^\star$ que maximice el valor de MSD. Esta métrica ha sido ampliamente usada en el contexto de la localización competitiva de instalaciones [8] y la selección de miembros de jurado [14].

La segunda medida de diversidad considerada, llamada *Max-Min Diversity (MMD)*, es evaluada como la distancia mínima entre elementos en S (ver la Ecuación 3).

$$MMD(S) = \min_{\substack{i,j \in S \\ i<j}} d_{ij} \tag{3}$$

De nuevo, el objetivo es encontrar una solución $S^\star$ que maximice el MMD. Esta métrica ha sido estudiada en problemas relacionados con la localización de instalaciones nocivas [5].

Max-MinSum Diversity (MMSD) es la tercera métrica de diversidad considerada en este trabajo. Para cada elemento seleccionado $i \in S$, se calcula la suma de distancias desde i a los elementos en $S \setminus \{i\}$, devolviendo el valor mínimo obtenido de entre todas las posibles sumas (ver la Ecuación 4).

$$MMSD(S) = \min_{i \in S} \sum_{j \in (S \setminus \{i\})} d_{ij} \tag{4}$$

El objetivo en este caso es maximizar el valor de MMSD. Esta métrica ha sido usada tradicionalmente en problemas relacionados con la equidad o en la selección de grupos homogéneos [3].

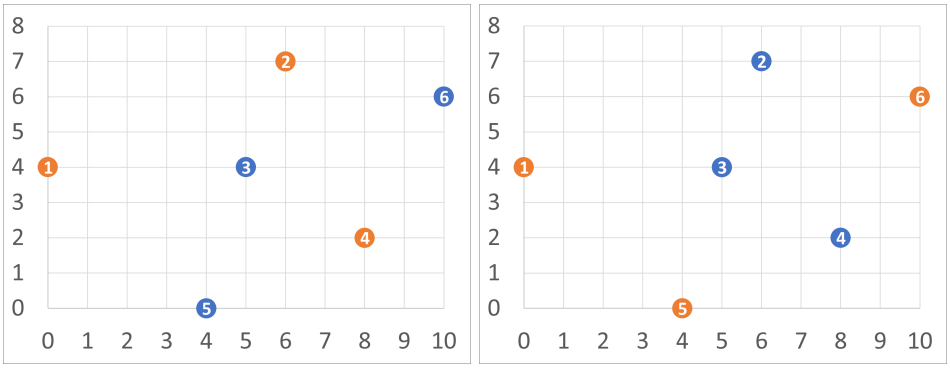
La cuarta medida de diversidad considerada en este trabajo es llamada *Minimum Differential Dispersion (MDD)*. Para evaluar esta medida, se utiliza un método similar al usado para evaluar el MMSD. Para cada elemento seleccionado $i \in S$, se calcula la suma de distancias al resto de elementos en $S \setminus \{i\}$, evaluando el MDD como la diferencia entre el máximo y el mínimo valor obtenido entre todas las posibles sumas (ver la Ecuación 5). El objetivo de esta medida es, por tanto, encontrar el valor mínimo del MDD.

$$MDD(S) = \max_{i \in S} \sum_{j \in (S \setminus \{i\})} d_{ij} - \min_{i \in S} \sum_{j \in (S \setminus \{i\})} d_{ij} \tag{5}$$

La quinta y última medida de diversidad considerada en este trabajo se denomina *Minimum p-Center Diversity (MPCD)*. Evaluar esta medida requiere el conjunto de elementos seleccionados S y también el conjunto de elementos no seleccionados $N \setminus S$. Para cada elemento no seleccionado $i \in N \setminus S$, se calcula la distancia a su elemento seleccionado más cercano $j \in S$, devolviendo la mayor distancia encontrada (ver la Ecuación 6).

$$MPCD(S) = \max_{i \in (N \setminus S)} \left\{ \min_{j \in S} d_{ij} \right\} \tag{6}$$

En este caso, el objetivo es minimizar el valor del MPCD. Este problema surge en el contexto de localización de instalaciones, donde el conjunto de elementos seleccionados hace referencia a las instalaciones que serían abiertas, mientras que el conjunto de no seleccionados representa a los puntos de demanda que dependen de los servicios de las instalaciones abiertas.



(a) Solución $S_1 = \{1, 2, 4\}$. (b) Solución $S_2 = \{1, 5, 6\}$.

Figura 1: Dos posibles soluciones para una instancia con 6 elementos disponibles de los cuales hay que seleccionar 3.

A continuación, se muestra la evaluación de estas métricas con un ejemplo gráfico. La Figura 1 muestra una instancia de ejemplo con $N = \{1, 2, 3, 4, 5, 6\}$ y $p = 3$. Las Figuras 1a y 1b ilustran dos posibles soluciones $S_1 = \{1, 2, 4\}$ y $S_2 = \{1, 5, 6\}$ para este ejemplo, respectivamente. Debemos destacar que los puntos están representados en un espacio bidimensional euclídeo para facilitar la evaluación de la distancia entre los puntos.

La Tabla I resume los resultados obtenidos en cada una de las métricas de diversidad en S_1 y S_2. Debemos destacar que el mejor resultado para cada métrica de diversidad se resalta en negrita.

Solución	MSD	MMD	MMSD	MDD	MPCD
S_1	20.34	5.39	12.09	**2.86**	**4.47**
S_2	**24.34**	**5.66**	**14.14**	4.54	**4.47**

Tabla I: Evaluación de las cinco métricas de diversidad propuestas sobre las soluciones presentadas en la Figura 1.

Los resultados presentados en la Tabla I confirman el comportamiento de un MOP. En concreto, S_1 obtiene mejor calidad que S_2 en la métrica MDD, mientras que S_2 supera a S_1 en MSD, MMD y MMSD. Finalmente, ambas soluciones presentan la misma calidad en la métrica MPCD. Por tanto, no es posible determinar que S_1 sea mejor que S_2 o viceversa, ya que son soluciones no dominadas.

III. Propuesta algorítmica

Este trabajo propone un *Greedy Randomized Adaptive Search Procedure (GRASP)* para resolver el MOMDP. GRASP fue presentado originalmente en [6] pero no fue formalmente definido hasta [7]. Es una metaheurística muy extendida que ha sido aplicada con éxito a muchos problemas de optimización complejos de una gran variedad de áreas [13, 18, 21]. GRASP sigue una estrategia multi-arranque formada por dos etapas diferentes: construcción y mejora. La primera consiste en una fase de construcción voraz aleatorizada para generar una solución inicial, mientras que la mejora es responsable de optimizar localmente la solución construida. El éxito de esta metodología se basa en el balance entre intensificación y diversificación en la generación de soluciones, que se controla por un parámetro de búsqueda llamado $\alpha \in [0, 1]$, que se describe en detalle en la Sección III-A.

III-A. Procedimiento constructivo

El procedimiento constructivo en el contexto de GRASP debe generar una solución inicial diversa y de alta calidad para comenzar la búsqueda. En lugar de usar un procedimiento completamente voraz, GRASP favorece la diversificación incluyendo aleatoriedad en este paso. El Algoritmo 1 presenta el pseudocódigo del procedimiento constructivo propuesto.

Algorithm 1 `GRASP Constructive`(N, p, α)

1: $i \leftarrow RND(N)$
2: $S \leftarrow \{i\}$
3: $CL \leftarrow N \setminus \{i\}$
4: **while** $|S| < p$ **do**
5: $\quad g_{\text{mín}} \leftarrow \text{mín}_{c \in CL}\, g(c)$
6: $\quad g_{\text{máx}} \leftarrow \text{máx}_{c \in CL}\, g(c)$
7: $\quad \mu \leftarrow g_{\text{máx}} - \alpha \cdot (g_{\text{máx}} - g_{\text{mín}})$
8: $\quad RCL \leftarrow \{c \in CL : g(c) \geq \mu\}$
9: $\quad i \leftarrow RND(RCL)$
10: $\quad CL \leftarrow CL \setminus \{i\}$
11: $\quad S \leftarrow S \cup \{i\}$
12: **end while**
13: **return** S

El procedimiento constructivo requiere tres parámetros de entrada: N, el conjunto de elementos candidatos para ser seleccionados; p, el número de elementos que deben ser seleccionados; y α, un número real en el rango $[0, 1]$ que controla el balance entre diversificación e intensificación. En el contexto de GRASP, el primer elemento que se incluye en la solución suele ser seleccionado de forma aleatoria del conjunto de disponibles N para favorecer la diversificación (paso 1). Este elemento es incluido en la solución que se está construyendo (paso 2), y se crea una lista de candidatos *CL* con todos los elementos menos i (paso 3). Hecho esto, el método añade de forma iterativa un nuevo elemento a la solución hasta

que se convierte en factible (pasos 4-12). En cada iteración, se calcula el valor mínimo ($g_{\text{mín}}$) y el máximo ($g_{\text{máx}}$) de una cierta función voraz (pasos 5-6). Esta función voraz $g(c)$ determina la contribución de cada elemento candidato c a la calidad de la solución en construcción.

Tras evaluar los candidatos, se calcula un umbral μ (paso 7) con el objetivo de construir una lista de candidatos restringida *RCL* (paso 8) con los elementos más prometedores. El umbral depende del valor del parámetro de entrada de búsqueda α, que controla la voracidad / aleatoriedad del método. Por un lado, si $\alpha = 0$, entonces $\mu = g_{\text{máx}}$, por lo que la *RCL* solo contiene aquellos elementos con el valor máximo de la función voraz, lo que resulta en un criterio totalmente voraz. Por otro lado, si $\alpha = 1$, entonces $\mu = g_{\text{mín}}$, lo que resulta en un criterio totalmente aleatorio ya que todos los elementos disponibles pueden entrar en *RCL*. El valor más adecuado para este parámetro de entrada se estudia en la Sección IV. Una vez la *RCL* ha sido construida con los elementos más prometedores, se escoge uno al azar para ser incluido en la solución en construcción (paso 9). Finalmente, *CL* se actualiza eliminado el elemento seleccionado (paso 10) e incluyéndolo en la solución (paso 11). El método termina cuándo se han seleccionado p elementos, devolviendo la solución S (paso 13).

Este esquema tradicional del constructivo GRASP debe ser adaptado a un problema de optimización multi-objetivo como MOMDP. En concreto, proponemos dos adaptaciones diferentes para construir soluciones factibles. La primera adaptación, llamada *Joint Alternate Evaluation of Objectives (JALEO)*, se centra en usar la métricas de diversidad presentadas en la Sección II como funciones voraces en cada construcción independientemente, generando el mismo número de soluciones para cada objetivo. La segunda adaptación, llamada *Mixed Objective Function Algorithm (MOFA)*, usa todas las métricas de diversidad como funciones voraces simultáneamente en cada construcción. Para esto, la función voraz varía de una métrica de diversidad a otra en cada iteración del procedimiento constructivo. En la Sección IV se compara el desempeño de ambas estrategias para construir soluciones iniciales. Finalmente, cabe destacar que, para cada solución construida, se evalúa la posibilidad de ser incluida en el conjunto de soluciones eficientes.

III-B. Búsqueda local

Las soluciones iniciales construidas con el método descrito en la Sección III-A no son necesariamente óptimos locales y, por tanto, pueden ser mejoradas para encontrar un óptimo local respecto a un cierto vecindario. El segundo paso del algoritmo GRASP es responsable de encontrar estos óptimos locales para cada solución construida.

El primer elemento requerido para definir una búsqueda local es un operador de movimiento. En el contexto del problema en consideración, proponemos un movimiento de intercambio el cual, dada una solución S y un conjunto de elementos N, elimina un elemento seleccionado $i \in S$ e incluye uno no seleccionado $j \in N \setminus S$. Cabe destacar que el movimiento de intercambio siempre resulta en una solución factible. En términos matemáticos,

$$Intercambio(S, i, j) = (S \setminus \{i\}) \cup \{j\}$$

Por tanto, el vecindario $\mathcal{N}(S)$ de una solución dada S se define como el conjunto de soluciones S' que pueden ser alcanzadas aplicando un solo intercambio a S. Formalmente,

$$\mathcal{N}(S) \leftarrow \{S' \leftarrow Intercambio(S, i, j) \quad \forall i \in S \wedge \forall j \in N \setminus S\}$$

La definición de una búsqueda local consiste en determinar cómo se recorre el vecindario en la búsqueda. En este trabajo se propone un enfoque basado en *first improvement*, que realiza el primer movimiento que conlleve una mejora, reduciendo los tiempos de cómputo. Debido a que se trata de un MOP, consideramos que se ha producido una mejora si la solución explorada entra en el conjunto de soluciones eficientes.

Al utilizar la estrategia *first improvement* es relevante determinar el orden en el cual el vecindario es explorado. Como una solución en $\mathcal{N}(S)$ se obtiene realizando un solo movimiento de *Intercambio*, el cual involucra a un elemento seleccionado $i \in S$, y a uno no seleccionado $j \in (N \setminus S)$, podemos determinar el orden el cual se exploran los elementos en S y, de la misma manera, con los elementos en $N \setminus S$. En particular, para cada i, podemos calcular la distancia mínima a los demás elementos seleccionados, i.e., $\text{mín}_{i' \in (S \setminus \{i\})} d_{i,i'}$. De esta forma, los elementos seleccionados se exploran en orden ascendente respecto a esta distancia, mientras que los no seleccionados se exploran en orden descendente. Este ordenamiento nos permite explorar primero los movimientos más prometedores.

IV. Resultados experimentales

Esta sección tiene dos objetivos principales: (i) ajustar los parámetros de entrada para encontrar la mejor configuración del algoritmo propuesto y (ii) realizar un análisis comparativo con el mejor método encontrado en el estado del arte para analizar el desempeño del algoritmo. Todos los algoritmos han sido implementados en Java SE 11, realizando los experimentos en un AMD EPYC 7282 (2.8 GHz) y 8 GB RAM.

El conjunto de instancias consideradas en este trabajo es el mismo que el presentado en el mejor método encontrado en la literatura [22], formado por el subconjunto GKD de la MDPLIB[1]. En concreto, se trata de 145 instancias que van desde 10 a 500 elementos donde el número de elementos a seleccionar va desde 2 hasta 50.

Al trabajar con problemas de optimización mono-objetivo, es fácil realizar una comparación entre dos o más algoritmos, ya que sólo es necesario comparar el valor de la función objetivo para cada algoritmo que se desee analizar. Sin embargo, en el contexto de la optimización multi-objetivo, como en el MOMDP, el resultado de un algoritmo es un conjunto de soluciones eficientes y, por tanto, es necesario definir métricas específicas para comparar conjuntos de soluciones eficientes.

[1] https://www.uv.es/rmarti/paper/mdp.html

En este trabajo se utilizan algunas métricas bien conocidas en la literatura de MOP: cobertura, hipervolumen, indicador ϵ, y distancia generacional inversa [12]. A continuación, se explica brevemente cada una de ellas.

La métrica cobertura, $C(R, \hat{E})$, calcula el número de soluciones del frente de aproximación en evaluación $\hat{E}$ que son dominadas por un frente de referencia R. Este frente de referencia se calcula como el conjunto de soluciones eficientes resultante de la unión de todos los frentes involucrados en la comparación. El hipervolumen, *HV*, evalúa el volumen en el espacio objetivo cubierto por el conjunto de soluciones eficientes en evaluación. El indicador ϵ, *EPS*, calcula cuál es la distancia más pequeña necesaria para transformar cada punto del conjunto de soluciones eficientes en evaluación al punto más cercano en el frente de referencia. La última métrica considerada es la distancia generacional inversa, *IGD+*, que consiste en invertir la métrica de distancia generacional con el objetivo de medir la distancia entre el conjunto de soluciones eficientes en evaluación con el frente de referencia. Finalmente, se ha incluido también el tamaño del conjunto de soluciones eficientes, $|\hat{E}|$ y el tiempo de cómputo requerido por cada algoritmo medido en segundos, *Tiempo (s)*. Todas las tablas presentadas en esta sección incluyen el valor medio de cada métrica respecto al conjunto de instancias consideradas, donde el mejor valor es resaltado en negrita.

Para empezar, es necesario ajustar los parámetros de entrada del algoritmo: α y el número de soluciones generadas. Para evitar un sobreajuste, se ha seleccionado un subconjunto de 20 instancias representativas (13 % del conjunto completo de instancias) para realizar todos los experimentos preliminares. En concreto, se han probado los valores de $\alpha = \{0{,}25, 0{,}50, 0{,}75, RND\}$, donde *RND* indica que el valor de α se selecciona aleatoriamente para cada construcción, resultando el constructivo JALEO con un valor de $\alpha =$ *RND* como la mejor opción para generar soluciones. Para el número de construcciones, se ha probado desde 100 hasta 1000, en intervalos de 100 construcciones, resultando 700 el mejor valor en cuanto al compromiso calidad-tiempo.

IV-A. Pruebas competitivas

El objetivo principal de esta sección es realizar una prueba competitiva entre el algoritmo propuesto y el mejor método previo encontrado en el estado del arte para analizar la eficiencia y eficacia el algoritmo GRASP propuesto (JALEO). El mejor método encontrado en la literatura es Vera et al. [22], donde se presenta un algoritmo evolutivo multi-objetivo (MOEA) para resolver el MOMDP. En concreto, los autores diseñan un *Non-Dominated Sorting Genetic Algorithm* (NSGA-II) [1] aprovechando la implementación ofrecida en la librería JMetal [4]. Para hacer una comparación justa, todos los experimentos han sido ejecutados en el mismo ordenador, utilizando la misma implementación de NSGA-II. Adicionalmente, se incluye el algoritmo evolutivo multi-objetivo basado en descomposición (MOEA/D) [23], para el cual se ha demostrado que es mejor que NSGA-II en rendimiento en trabajos recientes [11]. Ambos algoritmos han sido ejecutados con los parámetros determinados por los autores del artículo previo, es decir, 250000 evaluaciones y un tamño de población de 500. En este experimento final se ha usado el conjunto completo de 145 instancias descritas en la Sección IV. La Tabla II muestra los resultados finales de la prueba competitiva.

Algoritmo	$C(R, \hat{E})$	*HV*	*EPS*	*IGD+*	*Tiempo (s)*	$\lvert\hat{E}\rvert$
JALEO	**0.22**	**0.41**	**0.15**	**0.03**	**1.72**	**804.54**
NSGA-II	0.43	0.34	0.27	0.08	22.23	87.07
MOEA/D	0.26	0.35	0.28	0.12	25.29	20.57

Tabla II: Comparativa de GRASP, NSGA-II, y MOEA/D.

La primera conclusión de estos resultados es que MOEA/D es un método más competitivo que NSGA-II al considerar algoritmos evolutivos multi-objetivo, como sugería Li et al. [11]. Analizando primero la cobertura, JALEO obtiene el mejor valor (0.2223), seguido de cerca por MOEA/D. Los resultados obtenidos por NSGA-II indican que casi la mitad de las soluciones descubiertas por este algoritmo son dominadas por una o más soluciones del frente de referencia. En cuanto al hipervolumen, tanto MOEA/D como NSGA-II alcanzan valores similares (0.3495 *vs* 0.3511), siendo JALEO considerablemente mejor que ambos en este caso (0.4121). El mismo análisis es realizado al considerar el indicador ϵ, siendo de nuevo JALEO el mejor algoritmo. Además, JALEO es capaz de alcanzar el mejor valor para la métrica *IGD+*. Cabe mencionar que JALEO es capaz de incluir en el conjunto de soluciones eficientes una media de 804 soluciones, mientras que MOEA/D es capaz de incluir una media de solo 20 soluciones. Además, cabe destacar que JALEO es más de 10 veces más rápido que NSGA-II y MOEA/D, requiriendo, de media, menos de 2 segundos para resolver una instancia del MOMDP.

Finalmente, se realiza un test de Friedman no paramétrico para confirmar si hay diferencias estadísticas significativas entre los algoritmos comparados, utilizando las métricas de cobertura e hipervolumen. El valor-p obtenido menor de 0.0001 confirma esta hipótesis, resultando JALEO como el mejor del ranking (1.78), seguido de MOEA/D (1.95) y de NSGA-II (2.28). Analizando estos resultados, JALEO destaca como uno de los algoritmos más competitivos para resolver el MOMDP en cortos tiempos de cómputo. Finalmente, se aplica un test de Wilcoxon no paramétrico por pares para confirmar la superioridad de JALEO sobre MOEA/D. El valor-p menor de 0.0001 respalda esta hipótesis.

V. Conclusiones

En este artículo hemos presentado un enfoque original para resolver una variante multi-objetivo del problema de la máxima diversidad (MOMDP). Los problemas multi-objetivo se adaptan mejor a los problemas de la vida real, donde se considera normalmente más de un solo objetivo. Se proponen dos procedimientos constructivos diferentes siguiendo dos ideas completamente opuestas. Por un lado, JALEO genera soluciones optimizando cada una de las funciones voraces independientemente. Por otro lado, MOFA optimiza

simultáneamente todas las métricas de diversidad consideradas en cada construcción. El análisis experimental muestra que la estrategia seguida en JALEO es capaz de obtener mejores resultados. Se presenta además un procedimiento de búsqueda local eficiente que es capaz de limitar el vecindario explorado para encontrar un óptimo local que respeta las soluciones originalmente incluidas en el conjunto de soluciones eficientes. Se trata de una adaptación original de la metodología GRASP para los problemas de optimización multi-objetivo. La rigurosa experimentación computacional está respaldada con test estadísticos no paramétricos que confirman la calidad de la propuesta, destacando JALEO como el mejor algoritmo metaheurístico para resolver el MOMDP.

Este esquema GRASP se puede aplicar a cualquier problema multi-objetivo de la literatura. El código fuente sobre el que se han ejecutado los experimentos de este artículo está disponible en https://grafo.etsii.urjc.es/momdp.

Referencias

[1] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE transactions on evolutionary computation*, vol. 6, no. 2, pp. 182–197, 2002.

[2] K. Dhir, F. Glover, and C.-C. Kuo, "Optimizing diversity for engineering management," in *Proceedings of Engineering Management Society Conference on Managing Projects in a Borderless World*. IEEE, 1993, pp. 23–26.

[3] A. Duarte, J. Sánchez-Oro, M. G. Resende, F. Glover, and R. Martí, "Greedy randomized adaptive search procedure with exterior path relinking for differential dispersion minimization," *Information Sciences*, vol. 296, pp. 46–60, 2015.

[4] J. J. Durillo and A. J. Nebro, "jmetal: A java framework for multi-objective optimization," *Advances in Engineering Software*, vol. 42, no. 10, pp. 760–771, 2011.

[5] E. Erkut and S. Neuman, "Analytical models for locating undesirable facilities," *European Journal of Operational Research*, vol. 40, no. 3, pp. 275–291, 1989.

[6] T. A. Feo and M. G. Resende, "A probabilistic heuristic for a computationally difficult set covering problem," *Operations Research Letters*, vol. 8, no. 2, pp. 67–71, 1989.

[7] T. A. Feo, M. G. C. Resende, and S. H. Smith, "A greedy randomized adaptive search procedure for maximum independent set," *Operations Research*, vol. 42, no. 5, pp. 860–878, 1994.

[8] F. Glover, C.-C. Kuo, and K. S. Dhir, "Heuristic algorithms for the maximum diversity problem," *Journal of information and Optimization Sciences*, vol. 19, no. 1, pp. 109–132, 1998.

[9] M. J. Kuby, "Programming models for facility dispersion: the p-dispersion and maxisum dispersion problems," *Mathematical and Computer Modelling*, vol. 10, no. 10, p. 792, 1988.

[10] C.-C. Kuo, F. Glover, and K. S. Dhir, "Analyzing and modeling the maximum diversity problem by zero-one programming," *Decision Sciences*, vol. 24, no. 6, pp. 1171–1185, 1993.

[11] H. Li, K. Deb, Q. Zhang, P. N. Suganthan, and L. Chen, "Comparison between MOEA/D and NSGA-III on a set of novel many and multi-objective benchmark problems with challenging difficulties," *Swarm and Evolutionary Computation*, vol. 46, pp. 104–117, 2019.

[12] M. Li and X. Yao, "Quality evaluation of solution sets in multiobjective optimisation: A survey," *ACM Computing Surveys (CSUR)*, vol. 52, no. 2, p. 26, 2019.

[13] A. López-Sánchez, J. Sánchez-Oro, and A. Hernández-Díaz, "Grasp and vns for solving the p-next center problem," *Computers & Operations Research*, vol. 104, pp. 295 – 303, 2019.

[14] M. Lozano, D. Molina, C. García *et al.*, "Iterated greedy for the maximum diversity problem," *European Journal of Operational Research*, vol. 214, no. 1, pp. 31–38, 2011.

[15] R. Martí and F. Sandoya, "Grasp and path relinking for the equitable dispersion problem," *Computers & Operations Research*, vol. 40, no. 12, pp. 3091–3099, 2013.

[16] R. Martí, M. Gallego, A. Duarte, and E. G. Pardo, "Heuristics and metaheuristics for the maximum diversity problem," *Journal of Heuristics*, vol. 19, no. 4, pp. 591–615, 2013.

[17] F. Parreño, R. Álvarez Valdés, and R. Martí, "Measuring diversity. a review and an empirical analysis," *European Journal of Operational Research*, vol. 289, no. 2, pp. 515–532, 2021.

[18] S. Pérez-Peló, J. Sánchez-Oro, and A. Duarte, "Finding weaknesses in networks using greedy randomized adaptive search procedure and path relinking," *Expert Systems*, vol. 37, no. 6, p. e12540, 2020.

[19] G. Qian, S. Sural, Y. Gu, and S. Pramanik, "Similarity between euclidean and cosine angle distance for nearest neighbor queries," in *Proceedings of the 2004 ACM symposium on Applied computing*. New York, NY, USA: Association for Computing Machinery, 2004.

[20] R. Real and J. M. Vargas, "The probabilistic basis of jaccard's index of similarity," *Systematic Biology*, vol. 45, no. 3, pp. 380–385, 1996.

[21] S. Sohrabi, K. Ziarati, and M. Keshtkaran, "A greedy randomized adaptive search procedure for the orienteering problem with hotel selection," *European Journal of Operational Research*, vol. 283, no. 2, pp. 426–440, 2020.

[22] K. Vera, F. Lopez-Pires, B. Baran, and F. Sandoya, "Multi-objective maximum diversity problem," in *2017 XLIII Latin American Computer Conference (CLEI)*, 2017, pp. 1–9.

[23] Q. Zhang and H. Li, "Moea/d: A multiobjective evolutionary algorithm based on decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 6, pp. 712–731, 2007.

XIV Congreso Español de Metaheurísticas, Algoritmos Evolutivos y Bioinspirados

Sesión especial:

Algoritmos Bioinspirados Profundos



Una Visión Panorámica de la Meta-Cooperación Profunda con Algoritmos Meméticos

Jhon Edgar Amaya
Lab. de Computación
de Alto Rendimiento (LCAR).
Univ. Nacional Exp. del Táchira (UNET)
Email: jedgar@unet.edu.ve

Carlos Cotta*†, Antonio J. Fernández-Leiva*†
*ITIS Software, Universidad de Málaga
†Dept. Lenguajes y Ciencias de la Computación,
ETSI Informática, Universidad de Málaga,
Campus de Teatinos, 29071 - Málaga, España.
Email: {ccottap,afdez}@lcc.uma.es

Pablo García-Sánchez
Dept. de Arquitectura
y Tecnología de Computadores
ETSIIT, Universidad de Granada
Email: pablogarcia@ugr.es

***Abstract*—Los algoritmos meméticos son técnicas que orquestan la interacción entre componentes algorítmicos basados en la población y en la trayectoria. En particular, algunos modelos meméticos pueden considerarse bajo este amplio prisma como un grupo de algoritmos de optimización básicos autónomos que interactúan entre sí de forma cooperativa para abordar un problema de optimización específico, con el objetivo de obtener mejores resultados que los algoritmos que lo constituyen por separado. Yendo un paso más allá de esta visión tradicional de los algoritmos de optimización cooperativos, este trabajo aborda la meta-cooperación profunda, es decir, el uso de algoritmos de optimización cooperativos en los que algunos componentes pueden ser a su vez métodos cooperativos en sí mismos, mostrando así una arquitectura algorítmica profunda. El objetivo de este trabajo es revisar la evidencia disponible sobre la utilidad de estas técnicas y por la que pueden considerarse como una alternativa eficiente a otras formas tradicionales de algoritmos cooperativos.**

***Index Terms*—Algoritmos Meméticos, Arquitectura Profunda, Meta-cooperación, Conmutación de Herramientas**

I. Introducción

Uno de los factores clave del éxito de los métodos metaheurísticos para tratar con Problemas de Optimización Combinatoria (POCs) complejos es el equilibrio entre la explotación y la exploración del espacio de búsqueda. Sin embargo, encontrar el mejor equilibrio entre estas dos facetas de la búsqueda es, de hecho, otro problema de optimización, y no es fácil de resolver. Una propuesta interesante encontrada en los métodos de optimización basados en poblaciones para abordar esta cuestión es el uso de poblaciones estructuradas, limitando así las interacciones entre los individuos y permitiendo una mejor exploración y explotación del espacio de búsqueda [1].

Por otro lado, la *hibridación* significa básicamente una unión sinérgica de diferentes componentes, cada uno de los cuales aporta diferentes características al proceso de búsqueda y/o proporciona mecanismos para la explotación del conocimiento del problema. Desde una perspectiva amplia, la hibridación puede incluir el uso de cualquier complemento específico del problema que aporte conocimiento del mismo (por ejemplo, decodificadores especializados, operadores de variación ad-hoc, etc.), mejorando así el proceso de búsqueda. Sin embargo, a menudo la hibridación se utiliza con una connotación más específica, a saber, la combinación de componentes algorítmicos de nivel superior (con frecuencia, técnicas que podrían utilizarse como métodos independientes, como otras metaheurísticas). Bajo este último prisma, la hibridación puede verse desde dos amplias perspectivas [2]: la integración y la cooperación.

La *integración* se refiere a la adición de una de las técnicas de optimización como componente de otro método de optimización, mientras que la *cooperación* se refiere generalmente a la creación de un mecanismo de intercambio de información entre métodos que se aplican uno tras otro o en paralelo.

El enfoque de optimización cooperativa equivale a la aplicación de varios componentes algorítmicos, cada uno de los cuales explora un paisaje de búsqueda específico a través de procesos de intensificación/diversificación —inherentes a la metaheurística utilizada— para obtener un mecanismo eficaz con capacidad de escapar de los mínimos locales mediante el intercambio de información sobre el espacio de búsqueda explorado.

Estos modelos de optimización colaborativa constituyen un marco muy adecuado para integrar diferentes técnicas de búsqueda: cada una de ellas puede explotar el conocimiento del problema de forma complementaria y tener una visión diferente del panorama de búsqueda.

Partiendo de trabajos anteriores sobre este tipo de modelos, en [3] discutimos la meta-cooperación profunda, es decir, los modelos cooperativos en los que al menos uno de sus componentes es un modelo cooperativo en sí mismo. Por lo tanto, unimos la noción de optimización cooperativa esbozada anteriormente con la idea de *metaheurística profunda* [4] dando como resultado un modelo unificado para construir potentes complejos algorítmicos para la optimización. Si bien es cierto que se puede interpretar desde diferentes ángulos, la meta-cooperación encaja de forma natural en el amplio paradigma memético, ya que constituye un marco para ordenar la interoperación entre diferentes métodos, ya sea basados en poblaciones o en búsqueda local, con el objetivo de tener efectos sinérgicos en virtud de las capacidades de explotación/exploración adecuadamente diversificadas de las técnicas implicadas.

En esta línea, este trabajo resume el estado de la cuestión en el ámbito de las arquitecturas meméticas profundas, tomando como base los avances obtenidos en [3], donde se propuso un

esquema básico que puede ser fácilmente instanciado en un número de algoritmos diferentes, y proporciona evidencia de que los algoritmos meta-cooperativos profundos son métodos de optimización efectivos para tratar problemas combinatorios, capaces de superar a los métodos cooperativos superficiales.

II. Estado de la cuestión

Esta sección está dedicada a discutir algunos conceptos relacionados con los modelos cooperativos.

A. Modelos cooperativos

En general, los algoritmos cooperativos pueden clasificarse desde diferentes puntos de vista; por ejemplo, [5] propuso una división basada en dos enfoques: considerando la diversidad de los algoritmos subyacentes en el modelo cooperativo o por el nivel de cooperación entre ellos.

También, desde otro punto de vista, se han discutido dos modelos principales para implementar esquemas de cooperación [6]: (a) el enfoque colaborativo, en el que la cooperación consiste en un conjunto de programas autónomos, normalmente llamados agentes, que contienen métodos de solución particulares y que se comunican entre ellos para compartir información, y (b) el enfoque integrador que básicamente combina estos elementos autónomos en una estrategia simple y unificada para manejar un problema. Cada agente, en teoría, debería ser capaz de abordar el problema de forma independiente. A continuación, describimos algunos de los trabajos que han abordado ambos enfoques.

B. Enfoques Colaborativos

Uno de los primeros ejemplos de algoritmos colaborativos es el enfoque ”ir con los ganadores” de Aldous y Vazirani [7], en el que se utilizan sucesivas épocas de sincronización para centrar la búsqueda de una colección de algoritmos basados en trayectorias que de otro modo serían independientes.

Gallardo et al. [8] propusieron el uso entrelazado de un algoritmo truncado de branch-and-bound con algoritmos evolutivos y meméticos: los primeros identificarían partes prometedoras del espacio de búsqueda y pasarían la información a los segundos, que la utilizarían para inicializar su población y explorar dichas regiones.

Aquí el proceso cooperativo se consigue combinando metaheurísticas individuales, implementadas por agentes individuales, e intercambiando soluciones entre ellos. Basándose en esta idea, Amaya et al. [9] describieron un modelo cooperativo que puede ser instanciado por diferentes topologías de comunicación para compartir candidatos a solución, y donde cada agente en el modelo cooperativo puede ser cargado con un algoritmo de búsqueda particular. Otro modelo basado en agentes autónomos fue descrito por Byrski et al. [10]. En este trabajo, los individuos son agentes activos de la población, y no estructuras de datos pasivas (como en un EA canónico), caracterizados por un genotipo y un meme que pueden cambiar durante el proceso evolutivo. Se probaron diferentes tipos de acciones, incluyendo una formulación matemática para analizar las interacciones entre agentes.

Enfoques más recientes, como la búsqueda colaborativa multiagente (MACS) [11] también se centran en la optimización multiobjetivo, basada en el movimiento de los agentes a diferentes regiones dominantes del espacio de búsqueda, y en el uso de pesos para cada función objetivo. Asimismo, Fernández Leiva y Gutiérrez Fuentes [12] han propuesto recientemente un modelo algorítmico para distribuir un número de Algoritmos Meméticos (interactivos/proactivos) centrados en el usuario (UcMAs) que actúan como agentes independientes y, eventualmente, se sincronizan para intercambiar información.

C. Enfoques integradores

En el caso de los enfoques integradores, se combinan diferentes métodos en una estrategia unificada. Por ejemplo, entre los primeros trabajos sobre este tema, Anandalingham y Friesz [13] proponen el uso de sistemas multiagente con diferentes funciones objetivo que trabajan de forma no cooperativa y siguen las mismas restricciones, pero existe un cierto tipo de jerarquía a la hora de influir en otros agentes.

Desde una perspectiva diferente, Crainic et al. [14] utilizaron un método con búsquedas múltiples como estrategia cooperativa. En concreto, se empleó la búsqueda de vecindario variable (VNS) como método de búsqueda local (LS) con un mecanismo de memoria central que gestionaba el intercambio de información; diferentes instancias de VNS cooperaban de forma asíncrona mediante el intercambio de información para resolver diferentes instancias del problema *p-median*.

D. Algoritmos meméticos y arquitecturas profundas

Una cuestión crucial en los algoritmos cooperativos es la gestión adecuada del equilibrio entre la exploración y la explotación del espacio de búsqueda. Un algoritmo cooperativo eficiente debe ocuparse de ello; en este sentido, éste es uno de los principios centrales de los algoritmos meméticos (AM) [15]. Así, mientras que en la denominada *visión restringida* de los AM se considera típicamente una combinación entre algoritmos basados en población (que pueden entenderse de naturaleza más diversificadora) y algoritmos basados en búsqueda local (más orientados a la intensificación), la *visión amplia* considera la inclusión de otras técnicas de diversificación/intensificación en forma de técnicas exactas [16], algoritmos de exploración de espacios de estados [17], etc.

Esta idea de orquestar una combinación adecuada de componentes algorítmicos, central en los AMs y en las técnicas relacionadas antes descritas, da lugar de forma natural a la noción de *metaheurística profunda*, cf. [4]: de forma análoga a los algoritmos de aprendizaje profundo [18], que presentan múltiples capas de procesamiento para aprender representaciones de datos con múltiples niveles de abstracción, los algoritmos bioinspirados profundos (y las metaheurísticas profundas en general) presentan múltiples capas interconectadas (o componentes, ya que en general no es necesario que haya una jerarquía de capas bien definida) que aportan diferentes características deseadas al proceso de búsqueda encapsulando las herramientas necesarias para abordar los

diferentes aspectos de la complejidad del problema (e incluso los del sustrato computacional), y cuya interacción optimiza el proceso de resolución. Existen diferentes razones por las que las metaheurísticas en general (y las técnicas basadas en poblaciones en particular) se prestan a las arquitecturas profundas [19]. Son paradigmas de búsqueda flexibles cuya plasticidad permite acomodar múltiples componentes, y cuya resiliencia y descentralización inherentes (una propiedad especialmente relevante de los algoritmos basados en poblaciones) favorecen su útil integración.

La flexibilidad y plasticidad son especialmente interesante habida cuenta de que no es extraño que se se incluya un componente de aprendizaje dentro de la dinámica del sistema. Por un lado, esta es una idea que ya está presente tanto en las hiperheurísticas [20] como en modelos adaptativos de algoritmos evolutivos [21]. Más cercano al paradigma más propiamente memético, existe la noción de "complejo memético" (*memeplex*) [22], como colección de memes interconectados en una red dinámica sujeta a aprendizaje. Esto enlaza de manera natural con la noción de *computación memética*, que gira en torno a la coordinación armónica de estructuras computacionales complejas compuestas por módulos que interactúan (es decir, memes) para la resolución de problemas.

III. Un marco para la meta-cooperación profunda

El modelo propuesto consiste en la agregación jerárquica en varios niveles de un conjunto de agentes algorítmicos –cargados con una metaheurística particular y soluciones iniciales aleatorias– que exploran concurrentemente diferentes partes del espacio de búsqueda. El proceso de búsqueda lo realiza cada agente durante un tiempo determinado (o número de evaluaciones), tras lo cual la mejor solución encontrada por cada agente se distribuye mediante el modelo específico de topología de comunicación cooperativa en cada nivel en el que se produce la cooperación anidada. Para formalizar esta idea, describamos un modelo básico de cooperación que luego puede ampliarse recursivamente en una determinada arquitectura profunda.

Sea $\mathbb{N}_n = \{1, \cdots, n\}$, y sea R una arquitectura con n agentes, donde cada agente a_i ($i \in \mathbb{N}_n$) en R contiene una metaheurística, que puede ser un método de búsqueda local, un método basado en la población, una técnica híbrida o cualquier otro esquema cooperativo.

Estos agentes participan en períodos de exploración aislados seguidos de una comunicación sincrónica. Denotamos como $cycles_{\max}$ el número máximo de ciclos de exploración/comunicación en este modelo cooperativo.

Ahora, sea S_i el conjunto de soluciones candidatas generadas por el agente a_i (es decir por la ejecución de la metaheurística[1] contenida en el agente i), y sea $\Lambda \subseteq \mathbb{N}_n \times \mathbb{N}_n$ la topología de comunicación[2] en R. El funcionamiento del modelo es como sigue: primero, todos los agentes comienzan con soluciones iniciales aleatorias. A continuación, se ejecuta el algoritmo a lo largo de un número máximo de ciclos, donde en cada ciclo se ejecuta la técnica de búsqueda incorporada en cada agente para actualizar su *pool* de soluciones asociadas. Las soluciones se suministran de un agente a otro, según la topología considerada. Un agente acepta una solución sólo si la solución recibida es mejor que la mejor solución almacenada por el agente en su correspondiente conjunto de soluciones candidatas.

En cuanto a la topología de comunicación, se pueden utilizar diferentes configuraciones. A lo largo de este trabajo se han considerado las siguientes: Anillo (*Ring*), Transmisión (*Broadcast*) y Al Azar (*Random*).

El algoritmo se ejecutará para un número máximo de evaluaciones $E_{\max}$ (valor que hay que fijar inicialmente) y para un número determinado de ciclos $C_{\max}$ (también fijado inicialmente). Obsérvese que cada ciclo de nuestro algoritmo cooperativo consume $E_C = E_{\max}/C_{\max}$ de evaluaciones. Este esfuerzo computacional se reparte entonces (de manera uniforme) entre todos los n agentes de manera que el método de búsqueda específico de cualquier agente consume E_C/n evaluaciones como máximo.

Por ejemplo, un algoritmo cooperativo basado en la topología aleatoria, compuesto por cuatro agentes cargados con un algoritmo evolutivo (EA), una técnica de búsqueda en vecindades variables (VNS), un algoritmo de búsqueda tabú (TS), y un algoritmo de optimización mediante colonias de hormigas (ACO) y con un número máximo de 10 ciclos se podría representar como $\Phi =$ 10Random(EA,VNS,TS,ACO). Extendiendo esta notación anterior, un algoritmo meta-cooperativo se representa como $\Phi^g = C_{\max}\Lambda(a_1, a_2, ..., a_n)$, donde g es el grado de meta-cooperación (o profundidad de la meta-cooperación). Una definición recursiva más precisa se encuentra en [3].

IV. Evidencia Empírica

La experimentación más extensa realizada sobre el modelo anteriormente descrito fue realizada en [3] sobre el problema de la conmutación de herramientas. Dicha experimentación se realizó sobre la base de los siguientes algoritmos básicos:

- Hill Climbing (HC) [23].
- Método de entropía cruzada con una única (CE) o con múltiples (CEM) funciones de masa probabilística [24].
- Algoritmo memético con búsqueda tabú (MATS) y con hill climbing (MAHC) [23]

Sobre la base de estos algoritmos, existía ya un modelo cooperativo de nivel-0 describible como 5Ring (MAHC,MATS,MAHC) presentado en [9] y que proporcionaba los mejores resultados conocidos sobre este problema (considerando un amplio conjunto de instancias definidas

[1] Por ejemplo, si el agente está cargado con un método de búsqueda local entonces sólo se generará una solución y se mantendrá de modo que $|S_i| = 1$, pero si el agente está cargado con un método basado en la población —por ejemplo, un algoritmo memético o genético clásico— entonces se generará un conjunto de soluciones candidatas de modo que $|S_i| \geqslant 1$.

[2] Es decir, si $(i, j) \in \Lambda$ entonces a_i enviará información al agente a_j en cada ciclo de sincronización dentro de la ejecución del algoritmo cooperativo como se describe más adelante en este trabajo.

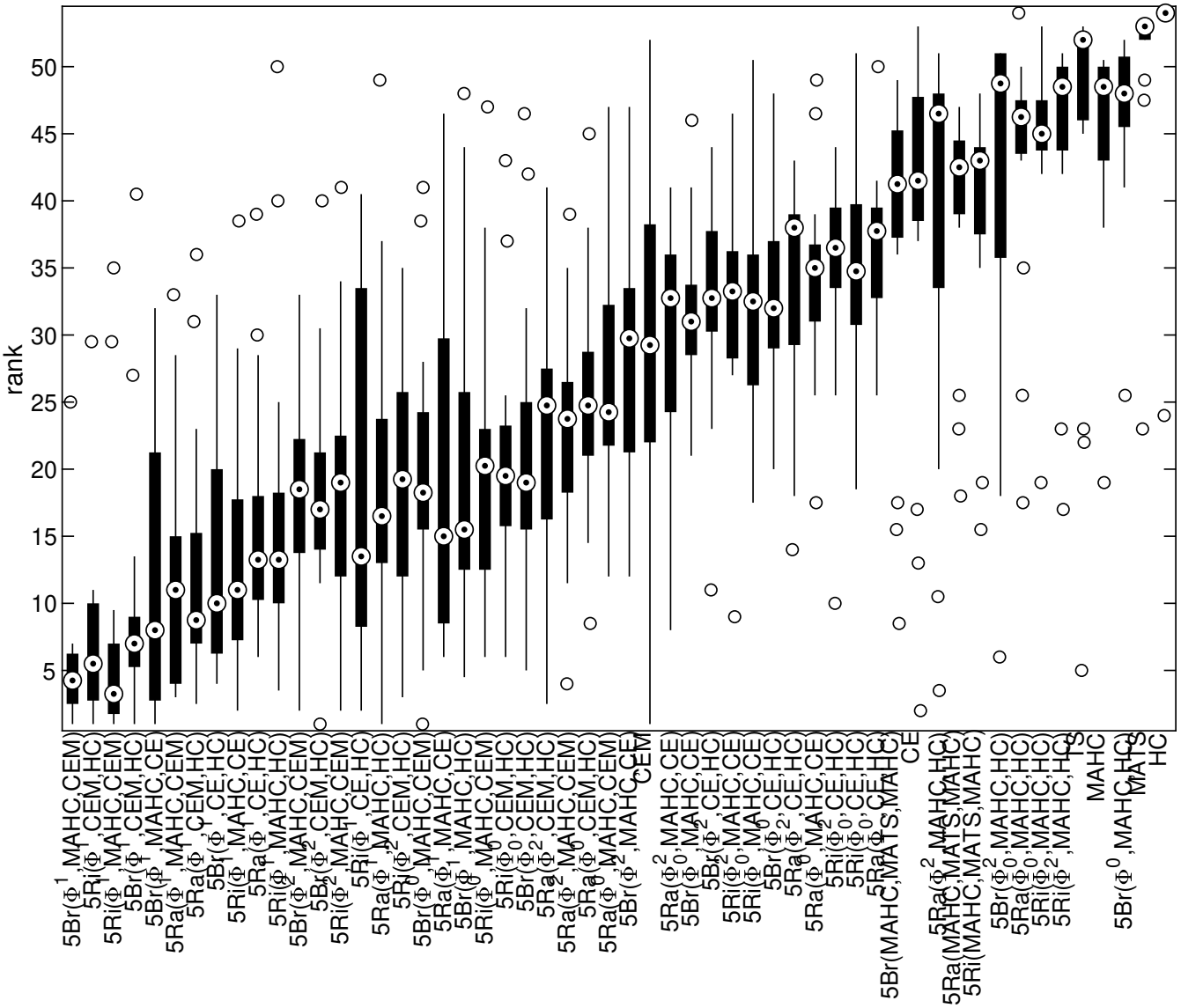


Fig. 1. Comparativa del rendimiento de los modelos considerados sobre el problema de la conmutación de herramientas. Para cada modelo se muestra la distribución de rangos (cómo se ordena en relación a los restantes algoritmos, donde 1 indica que es el mejor) sobre las diferentes instancias del problema considerado.

sobre la base de la literatura relacionada [25], [26]). Este algoritmo se tomó como primer algoritmo de control dentro de un esquema de experimentación incremental. Dicho esquema consiste en la construcción de diversos algoritmos meta-cooperativos sobre la base del algoritmo de control y la determinación del mejor de ellos. Este mejor algoritmo será el algoritmo de control en la siguiente etapa en la que se definirán nuevos modelos cooperativos con un nivel más de profundidad. De manera más precisa, los modelos meta-cooperativos considerados en [3] fueron:

- 5Λ(Φ^{g-1},CE,HC),
- 5Λ(Φ^{g-1},CEM,HC),
- 5Λ(Φ^{g-1},MAHC,CE),
- 5Λ(Φ^{g-1},MAHC,CEM)
- 5Λ(Φ^{g-1},MAHC,HC)

donde Φ^{g-1} es un modelo meta-cooperativo de nivel-$(g-1)$ y Λ es cualquiera de las tres topologías mencionadas anteriormente.

La experimentación con los modelos meta-cooperativos de nivel-1 indicó que los esquemas 5Λ(Φ^0,CEM,HC) y 5Λ(Φ^0,MAHC,CEM) presentaban el mejor rendimiento, con diferencias significativas sobre todos los demás en el caso de las topologías Broadcast y Ring. En particular 5Broadcast(Φ^0,CEM,HC) fue elegido como algoritmo de control Φ^1 al proporcionar el mejor rendimiento relativo.

Los resultados anteriores se trasladaron al caso de los meta-cooperativos de nivel-2, en los que los esquemas 5Λ(Φ^1,CEM,HC) y 5Λ(Φ^1,MAHC,CEM) proporcionaban el

mejor rendimiento (y en general, todos los esquemas de nivel-2 superaban a los de nivel-1). De manera más precisa, 5Broadcast(Φ^1,CEM,HC) era el mejor algoritmo con diferencia significativas sobre los esquemas que no usaban CEM o que usaban la topología Random. Este modelo fue elegido como algoritmo de control Φ^2.

Al llegar a los modelos meta-cooperativos de nivel-3 pudo apreciarse que el rendimiento no superaba a los modelos de nivel-2. Se había llegado por lo tanto al límite de complejidad asumible para el presupuesto computacional disponible (y para la escala de los problemas considerados). Evidentemente, el aumento de la profundidad tiene un peaje en el presupuesto computacional asignado a cada componente de la instancia particular considerada. Como resultado, éstos podrían no ser capaces de aprovechar un presupuesto reducido. Básicamente, esto es lo que ocurre en el caso de estudio descrito, en el que el esfuerzo de búsqueda está sobrediversificado a la luz del contexto computacional considerado. La Figura 1 muestra una comparativa global de todos los modelos considerados sobre todas las instancias.

A la luz de lo anteriormente descrito, está claro que hay dos factores que afectan al equilibrio entre intensificación y diversificación en el modelo: el número de agentes y la profundidad de la meta-cooperación. Como ya se ha indicado, para un presupuesto computacional fijo, tener un mayor número de agentes implica que cada uno de ellos recibe una menor cantidad de recursos para realizar la búsqueda. En algún momento, estos recursos pueden no ser suficientes para lograr una convergencia adecuada a regiones de alta calidad del espacio de búsqueda. Esto también es importante a la luz de la profundidad del modelo, ya que típicamente a mayor profundidad habrá un mayor número de algoritmos básicos. Otro factor importante es el hecho de que los modelos de mayor profundidad tendrán también mayor complejidad topológica, lo que puede suponer en general una difusión más lenta o gradual de información entre los algoritmos involucrados. Esto conlleva mayor diversificación, cosa que el diseñador del modelo puede intentar compensar mediante la elección de algoritmos de base más intensos.

V. Conclusiones

Los modelos de optimización meta-cooperativa profunda son un modelo apropiado para la integración de diferentes técnicas de búsqueda, cada una de ellas con una visión diferente del espacio de búsqueda. Así, combinando diferentes modelos de búsqueda y niveles de meta-cooperación, el proceso puede explotar una mayor capacidad para escapar de óptimos locales y centrarse en regiones prometedoras del espacio de búsqueda. Qué duda cabe que tal y como ocurre con los modelos de aprendizaje profundo, la interpretabilidad puede ser un problema. Al tratarse de modelos estructuralmente complejos, las decisiones de diseño pueden tener efectos no lineales en el rendimiento. Si bien esto puede poner de manifiesto la necesidad de realizar más estudios sobre la dinámica de la búsqueda, también apunta a la incorporación de propiedades de *self-star* [27] como medio para autogestionar el proceso de optimización. Además, está claro que este tipo de modelos sólo puede ser útil en escenarios en los que los algoritmos de búsqueda individuales no son lo suficientemente potentes por sí mismos, desperdiciando recursos computacionales en exploraciones improductivas.

El trabajo futuro en esta línea de trabajo puede desarrollarse en múltiples direcciones. El empleo de modelos de autogestión es sin duda uno de los objetivos más ambiciosos que pueden plantearse. Del mismo modo que un algoritmo multimemético puede descubrir complejos meméticos de interés, cabe plantearse si sería posible dotar de mecanismos de aprendizaje estructural que supervisen el proceso de búsqueda, ajustando tanto la composición como la disposición del modelo en respuesta al avance del proceso de búsqueda. Desde un punto de vista puramente computacional, es sin duda del máximo interés realizar una implementación distribuida de estos modelos, de forma que pueda explotarse la descentralización y asincronía inherentes a este tipo de modelos. Lógicamente, esto presenta desafíos no solo a nivel de desarrollo, sino también a nivel algorítmico, ya que un funcionamiento distribuido y asíncrono conlleva toda una serie de efectos sobre el proceso de búsqueda que deberán tenerse en cuenta a la hora de definir los modelos de manera adecuada para los problemas que se consideren.

Agradecimientos

Este trabajo está financiado en parte por el Ministerio Español de Economía y Competitividad (TIN2017-85727-C4-1-P, https://deepbio.wordpress.com/) y la Universidad de Málaga, Campus de Excelencia Internacional Andalucía Tech.

Referencias

[1] T. Y. Lim, "Structured population genetic algorithms: a literature survey," *Artif. Intell. Rev.*, vol. 41, no. 3, pp. 385–399, 2014.

[2] L. Jourdan, M. Basseur, and E. Talbi, "Hybridizing exact methods and metaheuristics: A taxonomy," *Eur. J. Oper. Res.*, vol. 199, no. 3, pp. 620–629, 2009.

[3] J. E. Amaya, C. Cotta, A. J. Fernández, and P. García-Sánchez, "Deep memetic models for combinatorial optimization problems: application to the tool switching problem," *Memetic Comput.*, vol. 12, no. 1, pp. 3–22, 2020. [Online]. Available: https://doi.org/10.1007/s12293-019-00294-1

[4] D. Camacho, R. Lara-Cabrera, J. J. Merelo Guervós, P. A. Castillo, C. Cotta, A. J. Fernández Leiva, F. Fernández de Vega, and F. Chávez de la O, "From ephemeral computing to deep bioinspired algorithms: New trends and applications," *Future Generation Comp. Syst.*, vol. 88, pp. 735–746, 2018.

[5] M. El-Abd and M. Kamel, "A taxonomy of cooperative search algorithms," in *Hybrid Metaheuristics, Second International Workshop, HM 2005, Barcelona, Spain, August 29-30, 2005, Proceedings*, ser. Lecture Notes in Computer Science, M. J. Blesa, C. Blum, A. Roli, and M. Sampels, Eds., vol. 3636. Springer, 2005, pp. 32–41.

[6] T. G. Crainic and M. Toulouse, "Explicit and emergent cooperation schemes for search algorithms," in *Learning and Intelligent Optimization 2007*, ser. Lecture Notes in Computer Science, V. Maniezzo, R. Battiti, and J. Watson, Eds., vol. 5313. Springer, 2007, pp. 95–109.

[7] D. Aldous and U. V. Vazirani, ""go with the winners" algorithms," in *35th Annual Symposium on Foundations of Computer Science, Santa Fe, New Mexico, USA, 20-22 November 1994*. IEEE Computer Society, 1994, pp. 492–501.

[8] J. E. Gallardo, C. Cotta, and A. J. Fernández, "On the hybridization of memetic algorithms with branch-and-bound techniques," *IEEE Trans. Systems, Man, and Cybernetics, Part B*, vol. 37, no. 1, pp. 77–83, 2007.

[9] J. E. Amaya, C. Cotta, and A. J. Fernández Leiva, "Memetic cooperative models for the tool switching problem," *Memet. Comput.*, vol. 3, no. 3, pp. 199–216, 2011.

[10] A. Byrski, R. Schaefer, M. Smolka, and C. Cotta, "Asymptotic guarantee of success for multi-agent memetic systems," *Bulletin of the Polish Academy of Sciences Technical Sciences*, vol. 61, no. 1, pp. 257 – 278, 2013.

[11] M. Vasile and L. A. Ricciardi, "Multi agent collaborative search," in *NEO 2015 - Results of the Numerical and Evolutionary Optimization Workshop NEO 2015 held at September 23-25 2015 in Tijuana, Mexico*, ser. Studies in Computational Intelligence, O. Schütze, L. Trujillo, P. Legrand, and Y. Maldonado, Eds., vol. 663. Springer, 2017, pp. 223–252.

[12] A. J. Fernández-Leiva and Á. Gutiérrez-Fuentes, "On distributed user-centric memetic algorithms," *Soft Computing*, Feb 2018.

[13] G. Anandalingam and T. L. Friesz, "Hierarchical optimization: An introduction," *Annals OR*, vol. 34, pp. 1–11, 1992.

[14] T. G. Crainic, M. Gendreau, P. Hansen, and N. Mladenovic, "Cooperative parallel variable neighborhood search for the *p*-median," *J. Heuristics*, vol. 10, no. 3, pp. 293–314, 2004.

[15] F. Neri and C. Cotta, "Memetic algorithms and memetic computing optimization: A literature review," *Swarm Evol. Comput.*, vol. 2, pp. 1–14, 2012.

[16] C. Cotta, A. J. F. Leiva, and J. E. Gallardo, "Memetic algorithms and complete techniques," in *Handbook of Memetic Algorithms*, ser. Studies in Computational Intelligence, F. Neri, C. Cotta, and P. Moscato, Eds. Springer, 2012, vol. 379, pp. 189–200.

[17] C. Cotta, "Protein structure prediction using evolutionary algorithms hybridized with backtracking," in *Artificial Neural Nets Problem Solving Methods*, ser. Lecture Notes in Computer Science, J. Mira and J. R. Álvarez, Eds., vol. 2687. Springer, 2003, pp. 321–328.

[18] Z. Cui, F. Xue, X. Cai, Y. Cao, G. Wang, and J. Chen, "Detection of malicious code variants based on deep learning," *IEEE Trans. Industrial Informatics*, vol. 14, no. 7, pp. 3187–3196, 2018.

[19] G. Wang, X. Cai, Z. Cui, G. Min, and J. Chen, "High performance computing for cyber physical social systems by using evolutionary multi-objective optimization algorithm," *IEEE Transactions on Emerging Topics in Computing*, 2019.

[20] K. Chakhlevitch and P. Cowling, "Hyperheuristics: Recent developments," in *Adaptive and Multilevel Metaheuristics*, ser. Studies in Computational Intelligence, C. Cotta, M. Sevaux, and K. Sörensen, Eds. Berlin Heidelberg: Springer-Verlag, 2008, vol. 136, pp. 3–29.

[21] L. Davis, *Handbook of Genetic Algorithms*. New York: Van Nostrand Reinhold, 1991.

[22] X. Chen and Y. S. Ong, "A conceptual modeling of meme complexes in stochastic search," *IEEE Transactions on Systems, Man and Cybernetics, Part C: Applications and Reviews*, vol. 42, no. 5, pp. 612–625, 2012.

[23] J. E. Amaya, C. Cotta, and A. J. Fernández Leiva, "Solving the tool switching problem with memetic algorithms," *AI EDAM*, vol. 26, no. 2, pp. 221–235, 2012.

[24] ——, "Cross entropy-based memetic algorithms: An application study over the tool switching problem," *Int. J. Comput. Intell. Syst.*, vol. 6, no. 3, pp. 559–584, 2013.

[25] M. A. Al-Fawzan and K. S. Al-Sultan, "A tabu search based algorithm for minimizing the number of tool switches on a flexible machine," *Comput. Ind. Eng.*, vol. 44, no. 1, pp. 35–47, 2003.

[26] B.-H. Zhou, L.-F. Xi, and Y.-S. Cao, "A beam-search-based algorithm for the tool switching problem on a flexible machine," *Int. J. Adv. Manuf. Technol.*, vol. 25, no. 9, pp. 876–882, May 2005.

[27] A. Berns and S. Ghosh, "Dissecting self-$\star$ properties," in *Third IEEE International Conference on Self-Adaptive and Self-Organizing Systems - SASO 2009*. San Francisco, CA: IEEE Press, 2009, pp. 10–19.

En busca de algoritmos genéticos eficientes desde el punto de vista energético

Jorge Alvarado
Grupo GEA
Universidad de Extremadura
C. Santa Teresa de Jornet, 38
06800, Mérida, España
jorgealvaradodiaz@gmail.com

Francisco Chávez
Ing. en Sistemas Informáticos y Telemáticos
C. Santa Teresa de Jornet, 38
06800, Mérida, España
fchavez@unex.es

Pedro A. Castillo
Dto de Arquitectura de Computadores
y Tecnología de Computadores
ETSIIT y CITIC, University of Granada
Granada, España
pacv@ugr.es

Josefa Díaz, Francisco Fernández de Vega
Arquitectura de Computadores
Universidad de Extremadura
C. Santa Teresa de Jornet, 38
06800, Mérida, España
{mjdiaz, fcofdez}@unex.es

***Resumen*—Cuando se aborda un problema de optimización con Algoritmos Evolutivos, se tienen en cuenta dos medidas principales para entender su rendimiento: la calidad de la solución y el tiempo de cálculo. Ambos valores se utilizan para comparar el rendimiento de diferentes versiones de un algoritmo, ajuste de parámetros del algoritmo o incluso para comparar los AE con otras heurísticas disponibles. Aunque normalmente la calidad de las soluciones y el tiempo de ejecución suelen ser las principales características de los algoritmos, recientemente una nueva tendencia en informática intenta contextualizarlas bajo una nueva perspectiva: el consumo de energía.**

Este trabajo presenta un análisis preliminar del algoritmo genético estándar, utilizando dos problemas bien conocidos y considerando la calidad de la solución, el tiempo de cómputo y el consumo de energía cuando se utilizan dispositivos alimentados por baterías para ejecutarlos. Los resultados muestran que algunos de los principales parámetros del algoritmo tienen un impacto en el consumo instantáneo de energía -que se aleja del comportamiento esperado- y, por tanto, en la cantidad de energía necesaria para ejecutar el algoritmo. Aunque todavía estamos lejos de encontrar la manera de diseñar Algoritmo Evolutivos energéticamente eficientes, creemos que los resultados abren una nueva perspectiva que nos permitirá alcanzar este objetivo en el futuro.

***Index Terms*—Consumo energético, Eficiencia de los algoritmos**

I. Introducción

Cuando se analizan los Algoritmos Evolutivos (AE) para conocer su rendimiento, los investigadores consideran tanto la calidad de la solución obtenida como el tiempo necesario para obtener dicha solución. Aunque se han desarrollado versiones paralelas de los algoritmos y se dispone de una gran cantidad de modelos y tecnologías de hardware [1], los investigadores siguen refugiándose en versiones secuenciales si el problema se puede resolver en poco tiempo (de minutos a horas). Por lo tanto, a partir de ahora nos referiremos a la versión secuencial del Algoritmo Genético (AG), aunque la discusión se podría adaptar a los paralelos y distribuidos.

Desde los inicios de los AGs, descritos por J. Holland en [2], los investigadores los han ejecutado sobre cualquier dispositivo que les permita obtener soluciones de calidad lo antes posible. Sin embargo, el consumo de energía del algoritmo no se ha considerado de interés, aunque sí lo han hecho otras áreas en el ámbito de la optimización [3]–[6]. Los investigadores han comprendido las posibilidades de los AEs para optimizar el consumo de energía en otras áreas [7]. Recientemente, algunos trabajos de revisión han incluido el consumo de energía como tema de interés a estudiar cuando se analiza el comportamiento de los AEs [8], particularmente cuando se utilizan pequeños dispositivos alimentados por baterías para ejecutar este tipo de metaheurísticas.

En los últimos años, se han publicado los primeros estudios sobre el comportamiento del consumo de energía en los algoritmos evolutivos [9], que incluyen un análisis preliminar de consumo de energía asociado a diferentes plataformas de hardware [10]. Este estudio es especialmente relevante cuando se utilizan dispositivos portátiles alimentados por baterías, por razones obvias. Sin embargo, hasta donde sabemos, no se ha presentado ningún estudio específico que analice el impacto de la configuración del algoritmo en la energía consumida para alcanzar una solución.

Así pues, este trabajo presenta por primera vez un análisis de este tipo para los AGs. Aunque los resultados son preliminares, consideramos que allanan el camino hacia una mejor comprensión del algoritmo bajo esta nueva perspectiva, que permitirá en el futuro el diseño de AGs más eficientes energéticamente.

El resto del artículo está organizado de la siguiente manera: la sección II presenta la motivación de este análisis y una discusión sobre los resultados esperados. La sección III describe la metodología y problemas seleccionados. La sección IV muestra los resultados obtenidos, y finalmente la sección V presenta las conclusiones y el trabajo futuro.

II. Considerando el consumo de energía en los AGs

Como se ha mencionado anteriormente, estamos interesados en analizar la huella de consumo energético de los algoritmos evolutivos, con especial atención a los dispositivos alimentados por baterías (ordenadores portátiles, tabletas, ipads, teléfonos móviles), que dependen de la energía disponible en ellas. Pero antes debemos entender por qué esta línea de investigación no se ha abordado antes, y por qué creemos que el análisis merece la pena.

Consideremos primero la versión estándar del Algoritmo Genético al abordar el problema *maxone*. En este caso, entre los principales parámetros del algoritmo se encuentran el tamaño del cromosoma, número de individuos de la población y número de generaciones.

Atendiendo al funcionamiento del algoritmo, se realizan "n" iteraciones del número de generaciones establecido y en cada generación se repiten una serie de operaciones estándar: evaluación del fitness, selección, cruce y mutación; cuanto mayor sea el tamaño de la población mayor será el número de repeticiones de las operaciones. Nuestro análisis considera una ejecución secuencial estándar del algoritmo. Esto significa que la CPU se dedicará al algoritmo, y las diferencias en el tiempo de ejecución debidas a la actividad del sistema operativo pueden descartarse si se utilizan datos de un gran número de ejecuciones para calcular los tiempos medios de ejecución. En realidad, la literatura apenas ha considerado la influencia de la actividad del sistema operativo en el tiempo de ejecución del algoritmo evolutivo. Probablemente, se consideren con modelos paralelos ejecutados en una única CPU, donde el sistema operativo debe gestionar los diferentes procesos.

II-A. Tamaño de la población y consumo de energía

Número de generaciones, tamaño de población y función de fitness son los componentes clave que influyen en el tiempo de ejecución de un AG estándar. Aunque, las operaciones de mutación pueden requerir un tiempo de cálculo diferente, creemos que los parámetros descritos anteriormente tienen una influencia mayor en el tiempo necesario para ejecutar el algoritmo. Así, ejecutar el algoritmo para N generaciones, tardará menos que $N+1$, y más que $N-1$. No consideramos aquí que la solución se encuentre antes de ese número de generaciones, lo que puede asegurarse incrementando la dificultad del problema (por ejemplo, aumentando el tamaño de los cromosomas en el problema maxone). Del mismo modo, si utilizamos I individuos en la población, el tiempo de ejecución será más corto que $I+1$, y más largo que $I-1$, dado que para cada individuo se debe calcular el fitness. Por último, ejecuciones similares con diferentes funciones fitness, el experimento con una función fitness que necesite más tiempo requerirá un tiempo de ejecución mayor. Se podría obtener una imagen más amplia del comportamiento del algoritmo con ejecuciones más largas para asegurar que el algoritmo encuentra el óptimo. Sin embargo, los problemas reales no suelen cumplir con esta idea y la ejecución se detiene cuando se alcanza el tiempo máximo de ejecución. Por ello, ajustamos la dificultad de los problemas para que no se encuentre una solución óptima en el tiempo asignado.

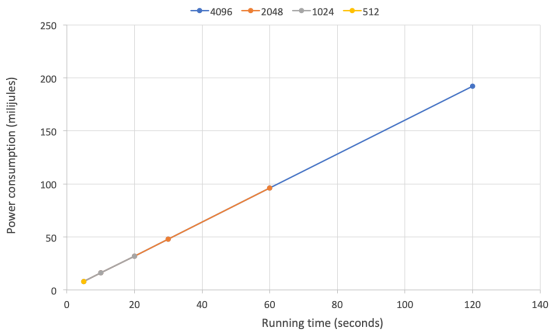


Figura 1. Consumo de energía esperado del AG con diferentes tamaños de población

Consideremos en primer lugar que al ejecutar el algoritmo, la CPU dedica exactamente el mismo esfuerzo independientemente de la operación que esté realizando. Esto significa que el consumo instantáneo de energía es el mismo a lo largo del experimento. Aunque no se ha realizado ningún estudio previo sobre esta cuestión para los AEs y aún no se ha descrito ningún interés en la literatura, podemos afirmar sin temor a equivocarnos que todo el mundo ha asumido que no hay ninguna diferencia real en la forma en que la CPU trabaja en un determinado experimento durante el tiempo que se ejecuta, independientemente de la operación específica que se realice en cada momento. La comunidad de AEs asume implícitamente que lo importante es el consumo total de energía de un experimento dado, y que podría calcularse fácilmente multiplicando el tiempo de ejecución y el consumo instantáneo de energía del procesador en un momento dado. Por tanto, nos centraremos aquí en la potencia total consumida por el AG.

Asumiendo las consideraciones previas, podríamos construir fácilmente un gráfico que muestre el consumo de energía estimado para diferentes valores de los parámetros en el AG. Por ejemplo, si una población de tamaño N (N individuos) consume una cantidad determinada de energía durante una ejecución, si ampliamos el tamaño de la población en una serie de experimentos, esperamos tener valores de consumo de energía proporcionales a la ampliación, dado que el algoritmo realizará tantas operaciones extra (evaluaciones de la solución, mutaciones, cruces...) proporcionales al nuevo número de individuos en la población. Mostramos el tipo de comportamiento esperado en la Figura 1.

En la figura se observa que el consumo es proporcional al valor del parámetro empleado. La única diferencia se debe al tiempo en completar el experimento: para un problema de dificultad suficiente cuanto mayor sea la población, mayor será el tiempo para completar el experimento. Por supuesto, se encontrarán mejores valores de fitness con poblaciones más grandes, pero ese no es el punto considerado aquí.

II-B. Dificultad del problema y tamaño del cromosoma

Una figura similar podría construirse utilizando la dificultad creciente del problema. Por ejemplo, cuando se aborda el problema *maxone* mediante AGs con tamaño de cromosoma creciente, que se correlaciona con la dificultad creciente, cuanto mayor sea el cromosoma mayor será el tiempo a evaluar. Asumiendo que no existen diferencias en la energía instantánea consumida en ninguno de los posibles experimentos a ejecutar, la energía total consumida sólo dependerá del tiempo, obteniendo una gráfica similar a 1 con diferentes tamaños de cromosomas.

Nuestra hipótesis es que este comportamiento esperado es lo que ha impedido a los investigadores tenerlo en cuenta al estudiar el comportamiento del AG: si el tiempo de ejecución y el consumo de energía son valores proporcionales, no hay razón para estudiar ambos. Una vez obtenido el tiempo de ejecución, el consumo de energía puede derivarse fácilmente.

Aunque así sea, recientemente se ha descrito que la eficiencia energética es clave para decidir la plataforma de hardware más eficiente para ejecutar un algoritmo [10].

En cualquier caso, ¿debemos seguir asumiendo que la energía y el tiempo son valores proporcionales de una única entidad sin pruebas experimentales? ¿Influyen algunos de los parámetros principales del AG de alguna manera en la energía necesaria para que el algoritmo se ejecute?

Creemos que este análisis es útil para confirmar o descartar la hipótesis asumida, y es lo que discutimos en las siguientes secciones. Además, en este análisis preliminar sólo nos centramos en el consumo de energía y no tenemos en cuenta la calidad de las soluciones encontradas. Sin embargo, creemos que los resultados presentados en este análisis preliminar muestran un nuevo camino para seguir mejorando los Algoritmos Genéticos, aunque esta vez con un punto de vista diferente.

III. Metodología

Se trata, por tanto, de estudiar si algunos de los principales parámetros del AG influyen en la potencia consumida por el procesador, y por tanto por la plataforma hardware, al ejecutar el algoritmo, o, por el contrario, no tienen ninguna influencia en ella.

III-A. Dispositivo con batería

En este primer estudio, se ha elegido una Lenovo Tablet Tab2, A10 - 70F con un SoC Mediatek MT8165, que es un modelo representativo bastante estándar entre las opciones disponibles.

Esta tablet contiene un MediaTek MT8165, SoC de 64 bits basado en ARM para dispositivos Android. El procesador de cuatro núcleos a 1,5 Ghz está fabricado en 28 nm y se basa en la arquitectura Cortex-A53. Además del núcleo de la CPU, integra una GPU ARM Mali-760 MP2 y un controlador de memoria LPDDR3 (32 bits, 800 MHz, 6,4 GB/s).

Tabla I
Parámetros & Valores analizados en ambos problemas

Problema maxone	
Tamaño población	32, 64, 128, 256,512, 1024 y 2048
Tamaño cromosomas	32, 64, 128, 256, 512, 1024, ... 32768
Problema Trap	
Tamaño población	32, 64, 128, 256, 512 y 1024

III-B. Medición del consumo de energía

Una vez seleccionado el dispositivo portátil, el consumo de energía del algoritmo se mide con una aplicación android, que permite analizar el consumo real de la batería de cualquier dispositivo portátil con Android ejecutando una versión Android del AG.

La medición se realiza con la aplicación desarrollada por la Universidad de Michigan llamada PowerTutor [12], que se compone de los siguientes elementos (i) un servicio web, (ii) un plug-in para la aplicación PowerTutor.

- Servicio Web: AG que se ejecuta como un sitio web, en el nuevo plug-in implementado para PowerTutor, obteniendo fácilmente su consumo de energía. Se utilizan dos enfoques; codificación desde cero y con la utilización de la librería NodEO [13]
- PowerTutor: plug-in que añade una nueva funcionalidad a PowerTutor. Dicho plug-in es capaz de ejecutar una tarea de computación y medir el consumo de energía de la CPU resultante de esa ejecución. El plug-in es el encargado de monitorizar el entorno web donde se ejecuta el AG, que debe ser codificado como un archivo JS y es ejecutado por el componente android Webview.

III-C. Problemas y parámetros analizados

Para este estudio preliminar se seleccionaron dos problemas de referencia de AG bien conocidos: *maxone* y función *trap* [11].

La idea es tener diferentes configuraciones de los principales parámetros del AG en experimentos largos y calcular la energía total consumida para cada uno de ellos. Se ejecuta la versión generacional del algoritmo, con un límite de tiempo máximo para la ejecución: 300 segundos; se abordó el problema maxone adaptando el tamaño de la población y la dificultad del problema (tamaño de los cromosomas), y luego, se probó la función trampa para confirmar algunas de las conclusiones extraídas.

La Tabla I resume los experimentos realizados para ambos problemas. Para cada uno de los valores de los parámetros se lanzaron 30 ejecuciones independientes por lo que se muestran los valores promediados. A pesar de la dificultad establecida para los problemas, algunas de las ejecuciones con poblaciones grandes encontraron la solución antes de los 300 segundos, por lo que en esos casos se calcularon los valores medios de las ejecuciones que alcanzaron ese paso de tiempo.

IV. Resultados

Comenzamos los experimentos utilizando el problema maxone con diferentes tamaños de cromosomas. En este problema

concreto, cuanto más largo sea el cromosoma, mayor será el tiempo para encontrar una solución, dado que la dificultad aumenta con el tamaño. Fijamos el tamaño de la población en 10, tratando de evitar tiempos cortos de ejecución, asegurando así la obtención de datos de consumo de energía para cada uno de los experimentos a lo largo de los 300 segundos de ejecución. Aunque existen otras posibilidades, creemos que este experimento aporta información relevante para el objetivo que perseguimos.

En la Figura 2, se observa que las diferencias entre las ejecuciones son escasas, aunque están presentes algunas diferencias en la parte alta de los valores de los parámetros: cuando los tamaños de los cromosomas son grandes. En cualquier caso, para apreciar mejor las diferencias en valores más cortos para este parámetro, se amplia la parte inferior del gráfico en la Figura3. De nuevo, se pueden apreciar algunas diferencias, aunque todavía estrechas.

Por lo tanto, vemos que el comportamiento considerando el tamaño del cromosoma no es muy diferente del esperado descrito en la sección anterior, aunque estas pequeñas diferencias encontradas podrían estudiarse más a fondo en el futuro.

Se realizan más experimentos utilizando diferentes valores para el tamaño de la población (ver Tabla I), y se fija el tamaño del cromosoma a un valor grande (16384 bits), que asegure que los experimentos se pueden realizar durante 300 segundos.

Los resultados se muestran en la Figura 4. Podemos notar que la figura es completamente diferente de la esperada (ver Figura 1): En lugar de todas las líneas superpuestas, el experimento muestra que existen diferencias importantes entre los tamaños de población. Especialmente relevantes son las diferencias cuando se emplean tamaños de población grandes, aunque para este problema específico, la energía consumida es menor en comparación con los tamaños de población pequeños, lo cual es un resultado sorprendente e intrigante.

Por lo tanto, decidimos centrarnos en los tamaños de población para el segundo problema, y así, lanzamos una serie de ejecuciones utilizando los tamaños de población que se muestran en la Tabla I. Los valores promedio de las 30 ejecuciones por tamaño de población se muestran en la Figura 5. En la Figura 5, se muestra de nuevo una progresión de líneas, cada una de ellas con un comportamiento de consumo de energía diferente, y una gradación de comportamientos casi perfecta: cuanto mayor es el tamaño de la población, mayor es la energía requerida. Sin embargo, la curva correspondiente al tamaño de población 512, no sigue la tendencia, y describe un comportamiento de consumo energético similar al tamaño de población 256. Hay que mencionar que con tamaños de población grandes, algunas de las ejecuciones encontraron la solución antes del final, por lo que los valores medios en los últimos pasos de tiempo incluyen un número menor de ejecuciones. Sea esta la razón o no, lo importante una vez más es que el comportamiento es diferente al esperado.

Los resultados descritos corresponden a los datos obtenidos cuando el AG se está ejecutando. También se ha medido lo que ocurre cuando la tableta está encendida y el algoritmo no se ejecuta. Por supuesto, el sistema operativo tendrá un papel,

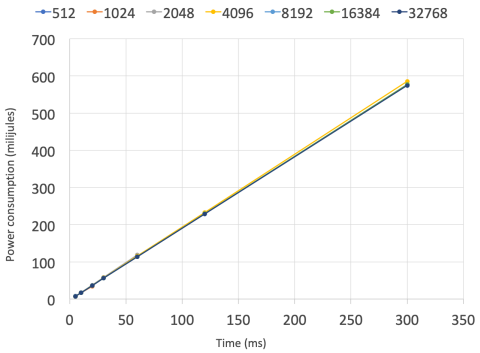


Figura 2. Consumo máximo de energía: Analizando diferentes tamaños de cromosomas. Tamaño de la población = 10

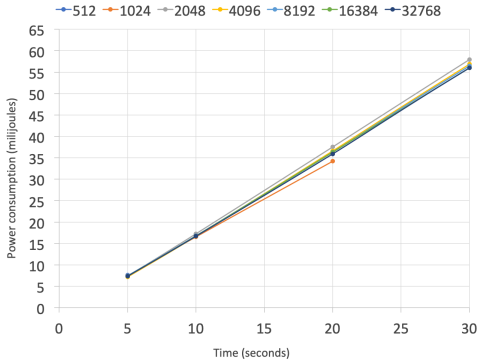


Figura 3. Consumo de energía de *maxone*: Analizando diferentes tamaños de cromosomas (zoom). Tamaño de la población = 10

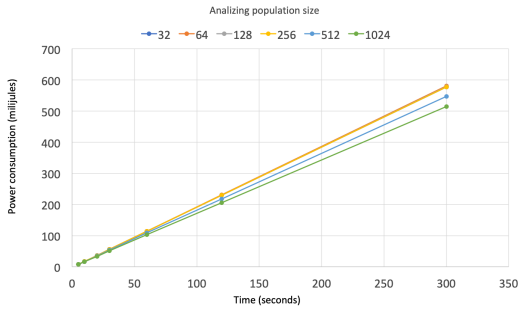


Figura 4. Consumo máximo de energía: análisis de diferentes tamaños de población (tamaño del cromosoma = 16384)

y algunos dispositivos pueden estar consumiendo energía, pero esta situación es similar a la que se da cuando cualquier programa se está ejecutando, por lo que medir la energía en esta situación ayudará a contextualizar los resultados anteriores.

Para una correcta comparación, se elimina del AG todas las operaciones a realizar, es decir el programa se ejecuta pero no realiza ninguna instrucción. La Figura 6 muestra la energía consumida a lo largo de treinta minutos, con diferentes pasos de tiempo. Los resultados son la media de treinta operaciones de reinicio de la tableta y la medición del consumo de energía

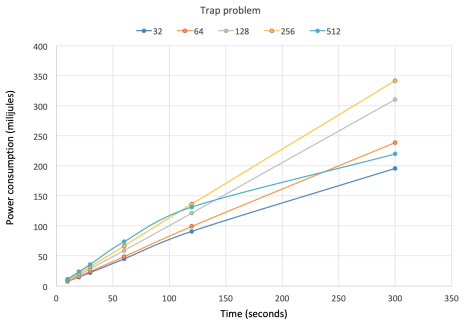


Figura 5. Consumo de energía de *Trap*: análisis de diferentes tamaños de población.

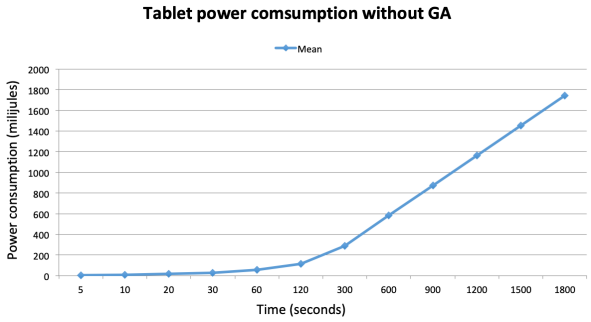


Figura 6. Energía consumida por el dispositivo de la tableta cuando el AG no está en funcionamiento

posterior. Si nos centramos en el valor de tiempo máximo asignado para los experimentos de AG (300), vemos que el consumo total de energía es inferior a 300. En cambio, si comprobamos de nuevo la Figura 4 los valores son muy superiores a 300: más de 500 milijulios para maxone. Para la función de Trap, las diferencias son menores, dado que sólo tenemos datos hasta 120 segundos.

La Tabla II presenta los datos del problema maxone para diferentes tamaños de cromosomas probados con 10 individuos en la población. Cada columna contiene la media del consumo de energía/desviación estándar de las 30 ejecuciones en diferentes pasos de tiempo. Los celdas vacías se deben a que los experimentos terminan antes de ese paso de tiempo. Se observa que cuando empleamos un tamaño de cromosoma de 4096 (y 2048 cuando hay datos disponibles), la energía consumida es mayor que con valores superiores a ese después de 30 segundos. Podemos comprobar, por ejemplo, el valor obtenido después de 300 segundos: 585,5 (DE = 4,9), mientras que para cromosomas más grandes la energía consumida es de unos 577. Esto confirma lo ya observado empíricamente en las figuras mostradas anteriormente: que existen diferencias significativas en los patrones de consumo de energía afectados por la configuración de los parámetros.

La Tabla III incluye información del problema maxone para diferentes tamaños de población con un tamaño de cromosoma de 16384 bits. En estos experimentos, cuando se utilizan poblaciones grandes (512 y 1024), la desviación estándar permite afirmar con seguridad que el consumo de energía es menor, comportamiento observado en la Figura 4 y que requiere más investigación en el futuro.

La Tabla IV detalla información similar para el problema Trap con diferentes tamaños de población combinados con un tamaño de cromosoma de 200 bits. Los resultados permiten de nuevo confiar en las diferencias encontradas. La última fila de las tres tablas anteriores presenta los valores medios y la desviación estándar calculados sobre el conjunto completo de pruebas para cada valor de tiempo abordado.

En definitiva, tras la serie de experimentos podemos confirmar que por primera vez hemos detectado que algunos de los valores de los parámetros principales, tienen un efecto sobre el consumo de energía del algoritmo; especialmente relevante es el tamaño de población empleado. Esto podría estar relacionado con los patrones de uso de la memoria, que pueden afectar a las operaciones de acceso a la caché. Sin embargo, se necesitan más experimentos para confirmar si este es un comportamiento general de los AGs, y que una configuración adecuada de los parámetros puede permitir diseñar AGs energéticamente eficientes.

V. Conclusiones

Este artículo presenta un primer análisis del consumo de energía de los Algoritmos Genéticos. Hasta donde sabemos, este análisis preliminar es el primero que estudia la influencia de algunos de los principales parámetros de los AG en la energía necesaria para ejecutar el algoritmo.

Aunque tradicionalmente la calidad de la aptitud y el tiempo hasta alcanzarla han sido las dos medidas principales para analizar el comportamiento de los AG, este trabajo introduce el análisis del consumo de energía al ejecutar los AG. Pocos estudios habían incluido anteriormente el consumo de energía como tema de interés. La razón principal es que el consumo de energía se consideraba de alguna manera proporcional al tiempo de ejecución en el pasado.

Se han abordado dos problemas: el problema de maxone y las funciones trap. Se han probado diferentes valores de tamaños de población y de cromosomas, y se ha comparado la energía necesaria para ejecutar el algoritmo durante 300 segundos con los resultados esperados teóricamente. En ambos problemas se han encontrado anomalías, que permiten ver una conexión entre los valores de los parámetros y el consumo de energía; además, en ocasiones el comportamiento muestra algunas anomalías -como un menor consumo con tamaños de población más grandes- que merecen una mayor investigación en el futuro.

Aunque los experimentos se han realizado en un dispositivo android alimentado con batería, esperamos ampliar el estudio a un conjunto más amplio de dispositivos, incluyendo ordenadores portátiles, raspberry, etc, de forma que podamos confirmar este comportamiento independientemente del hardware subyacente, y así poder diseñar Algoritmos Evolutivos más eficientes.

Tabla II
CONSUMO MÁXIMO DE ENERGÍA: ANALIZANDO DIFERENTES TAMAÑOS DE CROMOSOMAS. TAMAÑO DE LA POBLACIÓN = 10

	Time (s)						
Tamaño cromosoma	5	10	20	30	60	120	300
512	7.600/**0.44**						
1024	7.450/**0.36**	16.532/**0.49**	34.187/**0.61**				
2048	7.326/**0.53**	17.256/**0.62**	37.528/**1.04**	57.931/**1.51**	118.477/**1.60**		
4096	7.147/**0.51**	16.676/**0.66**	36.592/**0.94**	56.845/**1.01**	115.755/**1.55**	232.705/**3.27**	585.578/**4.90**
8192	7.456/**0.28**	16.791/**0.63**	36.315/**0.92**	56.435/**1.87**	113.782/**2.02**	229.485/**2.39**	577.026/**4.06**
16384	7.377/**0.36**	16.730/**0.62**	36.307/**0.85**	55.975/**1.57**	113.836/**1.84**	229.275/**2.65**	576.852/**3.73**
32768	7.344/**0.33**	16.678/**0.48**	35,897/**0.78**	55.975/**1.44**	113.472/**1.76**	228.783/**2.66**	574.544/**7.13**
Media/Desviación estándar	**7.386/0.425**	**16.777/0.623**	**36.354/1.195**	**56.632/1.652**	**115.041/2.562**	**230.040/3.128**	**578.256/6.504**

Tabla III
CONSUMO MÁXIMO DE ENERGÍA: ANALIZANDO DIFERENTES TAMAÑOS DE POBLACIÓN. TAMAÑO DEL CROMOSOMA = 16384

	Time (s)						
Tamaño población	5	10	20	30	60	120	300
32	7.334/**0.36**	16.487/**0.63**	36.056/**0.80**	55.761/**1.06**	113.828/**1.81**	229.618/**2.50**	576.326/**4.79**
64	7.295/**0.44**	16.755/**0.78**	35.767/**1.00**	55.994/**1.66**	113.972/**2.19**	230.513/**3.27**	580.024/**4.57**
128	7.353/**0.46**	16.791/**0.86**	36.383/**1.41**	56.016/**1.83**	113.738/**2.76**	229.958/**4.29**	577.435/**8.27**
256	7.401/**0.42**	16.902/**0.62**	36.089/**0.99**	55.552/**0.90**	114.025/**1.70**	229.713/**2.75**	576.974/**4.56**
512	7.415/**0.29**	16.528/**0.57**	34.473/**0.98**	53.178/**1.25**	108.981/**1.69**	217.755/**2.71**	546.043/**4.48**
1024	7.427/**0.23**	15.994/**0.57**	32.953/**0.90**	51.277/**0.99**	102.991/**1.57**	205.073/**1.91**	513.900/**2.95**
Media	**7.371/0.37**	**16.576/0.73**	**35.287/1.58**	**54.630/2.22**	**111.256/4.57**	**223.772/9.95**	**561.784/24.96**

Tabla IV
CONSUMO DE ENERGÍA DE LAS TRAMPAS: ANÁLISIS DE DIFERENTES TAMAÑOS DE POBLACIÓN. TAMAÑO DEL CROMOSOMA = 200

	Time (s)					
Tamaño población	10	20	30	60	120	300
32	7.63/**0.67**	14.95/**0.63**	22.36/**0.73**	45.14/**1.67**	91.18/**3.37**	195.95/**65.14**
64	8.06/**0.66**	16.18/**0.78**	24.45/**0.84**	48.90/**1.14**	99.35/**3.19**	238.89/**49.02**
128	9.3/**1.01**	18.74/**1.01**	28.94/**1.34**	59.18/**1.77**	121.50/**2.09**	310.71/**5.46**
256	10.47/**0.99**	21.16/**1.22**	32.25/**1.29**	66.73/**2.67**	136.64/**5.59**	342.17/**42.10**
512	11.48/**1.01**	23.79/**1.54**	35.96/**1.24**	73.78/**8.20**	131.64/**40.39**	220.00/**142.89**
Media/Desviación estándar	**10.413/1.668**	**21.398/3.353**	**31.645/9.360**	**105.098/16.990**	**116.368/27.049**	**260.954/91.138**

AGRADECIMIENTOS

Jorge Alvarado cuenta con el apoyo de la Comunidad de Madrid y los fondos FEDER bajo la proyecto Y2018/NMT-4668 (Micro-Stress- MAP-CM). Agradecemos el apoyo del Ministerio de Economía y Competitividad de España en el marco del proyecto TIN2017-85727-C4-{2,4}-P y PID2020-115570GB-C21. Junta de Extremadura, Consejería de Comercio y Economía, del Fondo Europeo de Desarrollo Regional, una forma de construir Europa, en el marco del proyecto IB16035 y de la Junta de Extremadura, proyecto GR15068.

REFERENCIAS

[1] Fernández de Vega, F., Hidalgo Pérez, J. I., and Lanchares, J., eds. "Parallel Architectures and Bioinspired Algorithms". Vol. 122. Heidelberg: Springer, 2012.

[2] Holland, J. H., .Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence."(1975).

[3] Ye, M. et al. "EECS: an energy efficient clustering scheme in wireless sensor networks."Performance, Computing, and Communications Conference, 2005. IPCCC 2005. 24th IEEE International. IEEE, 2005.

[4] Camilo, T., Carreto, C., Silva, J. S., & Boavida, F. (2006, September). An energy-efficient ant-based routing algorithm for wireless sensor networks. In International Workshop on Ant Colony Optimization and Swarm Intelligence (pp. 49-59). Springer, Berlin, Heidelberg.

[5] Albers, S., Energy-efficient algorithms. Communications of the ACM, 53(5):86–96, 2010.

[6] Heinzelman, W. R., Chandrakasan, A., & Balakrishnan, H. (2000, January). Energy-efficient communication protocol for wireless microsensor networks. In System sciences, 2000. Proceedings of the 33rd annual Hawaii international conference on (pp. 10-pp). IEEE.

[7] Gacto, M. J., Alcalá, R., & Herrera, F. (2012). A multi-objective evolutionary algorithm for an effective tuning of fuzzy logic controllers in heating, ventilating and air conditioning systems. Applied Intelligence, 36(2), 330-347.

[8] Camacho, D., Lara-Cabrera, R., Merelo-Guerv'os, J. J., Castillo, P. A., Cotta, C., Fernández-Leiva, A. J., Fernández de Vega & Chávez, F. (2018). From ephemeral computing to deep bioinspired algorithms: New trends and applications. Future Generation Computer Systems, 88, 735-746.

[9] Álvarez, J. D., Chávez, F., Castillo, P. A., García, J. A., Rodríguez, F. J., & Fernández de Vega, F. (2018). A Fuzzy Rule-Based System to Predict Energy Consumption of Genetic Programming Algorithms. Computer Science & Information Systems, 15(3).

[10] Fernández de Vega, F., Chávez, F., Díaz, J., García, J. A., Castillo, P. A., Merelo, J. J., & Cotta, C. (2016, September). A cross-platform assessment of energy consumption in evolutionary algorithms. In International Conference on Parallel Problem Solving from Nature (pp. 548-557). Springer, Cham.

[11] Deb, K., & Goldberg, D. E. (1993). Analyzing deception in trap functions. In Foundations of genetic algorithms (Vol. 2, pp. 93-108). Elsevier.

[12] Gordon, Mark, et al. PowerTutor: A power monitor for android-based mobile platforms. An Android Application, 2013.

[13] Merelo, Juan-Julián, et al.: NodEO, a multi-paradigm distributed evolutionary algorithm platform in JavaScript. Proceedings of the Companion Publication of the 2014 Annual Conference on Genetic and Evolutionary Computation. ACM, 2014. p. 1155-1162.

Clusterig Cosmológico: un enfoque del clustering gravitacional clásico inspidado en la estructura y dinámica del cosmos a gran escala

1º Aitor Castillo-López, 2º Javier Fumanal-Idocin, 3º Javier Fernández y 4º Humberto Bustince
Departamento de Estadística, Informática y Matemática, Universidad Pública de Navarra, (31006) Pamplona, Spain
aitor.castillo@unavarra.es

***Abstract*—En este trabajo proponemos un nuevo enfoque del algoritmo de clustering gravitacional basado en lo que Einstein consideró su "mayor error": la constante cosmológica. De manera similar al algoritmo de clustering gravitacional, nuestro enfoque está inspirado en principios y leyes del cosmos, y al igual que ocurre con la teoría de la relatividad de Einstein y la teoría de la gravedad de Newton, nuestro enfoque puede considerarse una generalización del agrupamiento gravitacional, donde, el algoritmo de clustering gravitacional se recupera como caso límite. Además, se desarrollan e implementan algunas mejoras que tienen como objetivo optimizar la cantidad de iteraciones finales, y de esta forma, se reduce el tiempo de ejecución tanto para el algoritmo original como para nuestra versión.**

***Index Terms*—cosmos, clustering, no supervisado, simulación, fuerza gravitacional**

I. Introducción

EL CLUSTERING, como parte del aprendizaje no supervisado, es de vital importancia para la minería de datos y la extracción de conocimiento a partir de estos. Conocer los datos y su posible estructura puede darnos información sobre el proceso subyacente de los datos en sí. Existe una gran cantidad de datos sin etiquetar accesibles a través de Internet y también es común encontrar información incompleta, faltante o mal recopilada.

Es por tanto de interés en la comunidad científica el análisis de los datos sin etiquetar y el por qué de que exista una amplia literatura orientada a realizar la extracción de conocimiento de los mismos. Concretamente, el clustering consiste en agrupar observaciones en diferentes grupos denominados "clusters", y estos pueden agruparse de acuerdo con algunos criterios, para maximizar la similitud dentro de dichas agrupaciones y minimizar la similitud entre estas. Dependiendo del algoritmo de agrupación, es posible que sea necesario especificar el número de clusters, así como otros parámetros relacionados con los datos. La extensa lista de algoritmos de clustering pueden dividirse en dos clases; sean algoritmos en los que se inicialmente se establece el número de clusters (o no-jerárquicos): k-means, k-medoids, hclust, pam, mona... Y en los que no es necesario establecer un número fijo de grupos de antemano (jerárquicos): agnes, diana... Además, el conocimiento y la influencia de diferentes ramas de la ciencia pueden inspirar a investigadores para modificar y basar algoritmos de clustering en reglas de la naturaleza tales como las biológicas o leyes físicas como es el algoritmo de clustering gravitacional [8].

La esencia del clustering gravitacional es imitar las leyes de la gravedad de Newton para atraer partículas (datos) entre sí y formar los grupos debido a la fuerza de atracción entre ellas. Esta fuerza de atracción hace que las partículas se acerquen unas a otras hasta que dos o más partículas estén lo suficientemente cerca como para fusionarlas todas en un único punto. Hasta este punto, casi todas las generalizaciones de este tipo de clustering jerárquico coinciden.

A partir de este punto, hay muchas formas diferentes de realizar el algoritmo. Es posible estudiar el papel de las masas en el algoritmo a través de funciones de superposición [1], así como cambiar la suma de las masas en colisión por otros valores como el maximo [4], etc. Pero también es posible estudiar cómo la naturaleza en el la fuerza subyacente entre las partículas puede hacer prosperar nuevos algoritmos de clustering como, por ejemplo, el enfoque de agrupamiento basado en dinámica molecular [3].

El modelo de clustering gravitacional original tiene ciertos problemas. Uno de ellos está relacionado con el hecho de que no siempre es correcto seleccionar la configuración más larga en tiempo como la más estable. Para los clusters en un entorno ruidoso esto no es cierto por el truncamiento del tiempo de vida de configuraciones concretas debido a la incorporación de los outliers los clusters. El segundo problema, el cual también se intenta abordar con este trabajo, es que el modelo original genera un problema de tipo "agujero negro". La formación de grupos de acuerdo con la fuerza de gravitación clásica produce objetos extremadamente masivos que impregnan el espacio de los datos con un campo tan fuerte que ignora las fuerzas entre otros objetos de baja masa. A estos puntos formados por una gran cantidad de datos bien se les podría denominar agujeros negros debido a su similitud con el término acuñado por John Wheeler [6] para referirse a los masivos cuerpos de los que ni los fotones pueden escapar. Por lo tanto, los datos tienden a ser atraídos por estos clusters más poblados o de alta densidad. Si en un conjunto de datos hay una gran diferencia entre las densidades de los clusters, un cluster densamente poblado puede convertirse en un agujero negro y atrapar otras partículas que no debiera (outliers) o incluso puede absorber otros clusters no tan masivos pero sí relevantes.

Volviendo a términos de física, el modelo gravitacional de Newton puede describir casi toda la dinámica en la escala del sistema solar. Pero, a la inversa, cuando uno intenta describir las trayectorias de los objetos en una escala mayor como el espacio interestelar o intergaláctico, las leyes clásicas fallan al intentar predecir dichos movimientos. Como analogía, este estudio pretende emular los clusters como galaxias y tratar el espacio de datos como un pequeño cosmos. Siendo así, al igual que en 1917 Einstein añadió la constante cosmológica a sus propias ecuaciones en un intento de unificar tanto la escala planetaria como la cosmológica [2], incluimos igualmente una fuerza de expansión a la fórmula clásica de Newton que puede, en cierto grado, corregir los problemas antes mencionados.

Finalmente, simplemente decir que a lo largo del trabajo y para acompañar las modificaciones teóricas propuestas, presentamos algunos ejemplos ilustrativos para mostrar la validez de nuestro enfoque. En la sección II explicamos el algoritmo original y su mecánica y en la sección III presentamos el fundamento y la formulación de nuestro nuevo enfoque. En la sección IV presentamos la experimentación y los resultados obtenidos al aplicar nuestro método en varios conjuntos de datos. Finalmente, en la sección V resumimos y concluimos con algunas observaciones.

II. Preliminares: Algoritmo de clustering gravitacional original

El algoritmo de clustering gravitacional se inspira en las leyes gravitacionales de Newton para el proceso de categorización de los datos. La mecánica se basa en mover las "partículas" que componen el dataset inicial entre sí de acuerdo con una relación gravitacional concreta, uniendo así unas partículas con otras a lo largo del tiempo para formar grupos que terminan por agruparse en un único punto (en teoría el centro de masas). Concretamente, para n partículas que denotaremos $p_1, p_2 ... p_n$ se encontrarán en las posiciones $s_1, s_2 ... s_n \in \mathbb{R}^N$ en un espacio N dimensional.

Este configuración de datos inicial se deja evolucionar siguiendo los pasos del Algoritmo1 como sistema. A lo largo de la vida de este, cada partícula del sistema corresponde a una categoría, siendo al principio tantas categorías como partículas y finalmente obteniendo una única categoría. La duración de todo el proceso se denota por T e implica lo siguiente:

- Desde $t_n = 0$ hasta t_{n-1} hay n (todas) las partículas.
- Desde t_{n-1} a t_{n-2} quedan $n-1$ partículas.
- ...
- De t_2 a $t_1 \equiv T$ quedan 2 partículas.

La vida relativa de una configuración con c clusters (categorías) se puede calcular como;

$$\frac{R_c = t_{c-1} - t_c}{T,}$$

donde la configuración más larga en el tiempo, interpretada como la más estable, determina el número de clústeres c.

Algoritmo 1: Clustering gravitacional de Wright

Entrada: Un dataset con n datos de N variable
Salida: Un dendrograma del proceso de unión
Inicialmente se asigna una masa $m_i = 1$ a cada partícula p_i donde $i \in [2, n]$;
Se escogen los parámetros positivos reales ϵ y δ;

- Se utiliza δ para determinar la longitud del paso de tiempo simulado entre iteraciones dt. Concretamente, en el tiempo transcurrido $[t, t + dt]$ la partícula más rápida se mueve δ.
- Si hay dos partículas a una distancia menor que ϵ, se unifican en una con la masa igual a la suma de las masas de ambas partículas en colisión y se establece la nueva posición en su centro de masas.

El tiempo se inicializa a $t = 0$;
mientras $n > 1$ **hacer**

1) En cada lapso temporal $[t, t + dt]$, para cada partícula p_i calculamos su función de influencia de movimiento:
$$g(i, t, dt) = \frac{1}{2} G \sum_{j \ neq i} \frac{m_i m_j}{m_i} \frac{s_j - s_i}{|s_j - s_i|^3} dt(t)^2 \quad (1)$$
donde G es una constante positiva. Nótese que el intervalo de tiempo dt no es una constante sino que depende de la iteración.
2) Para cada p_i, su nueva posición será:
$$s_i(t + dt) = s_i(t) + g(i, t, dt)$$
3) Elevamos t a $t + dt$.
4) Si dos partículas p_i y p_j donde $j \in [2, n]$, están a una distancia menor que ϵ, su unión se realiza como se explica arriba.
5) Se categorizan las partículas unídas bajo la misma etiqueta para construir el dendrograma.
6) Se reasigna n.

Una vez se detiene el bucle, se evalua el historial del sistema (dendrograma) para encontrar la configuración más estable en el tiempo.

III. Enfoque de la dinámica cosmológica

Como comentamos en la sección I, nuestro enfoque está inspirado en la constante cosmológica de Eistein. Es bien sabido por los astrónomos y cosmólogos que Einstein incluyó el parámetro Λ como un término en sus ecuaciones de campo porque sus ecuaciones no permitían, aparentemente, un universo estático: la gravedad haría que un universo que inicialmente estaba en equilibrio dinámico se contrajera [7].

De manera análoga, nosotros introducimos un factor Λ en la Eq. 1 del algoritmo de clustering gravitacional para tratar de maximizar la vida de una configuración estacionaria. No obstante, en primer lugar, se debe generalizar la notación la cual se explica brevemente en la siguiente subsección para abordar nuestra propuesta.

A. Generalización para algoritmos de simulación con propiedad de Markov basados en fuerzas isótropas

En la teoría de probabilidad es conocido como modelo o cadena de Márkov a un tipo de proceso estocástico discreto en el que la probabilidad de que ocurra un evento depende solamente del evento inmediatamente anterior. Esta característica de "falta de memoria" recibe el nombre de propiedad de Markov [5].

Si bien en nuestro caso las configuraciones que toman las partículas del sistema en cada paso no son estocásticas y están completamente determinadas por la fuerza percibida por cada una de ellas, la propiedad de Markov viene al dedo para describir todas aquellas simulaciones donde la configuración de un sistema en un tiempo concreto t está unicamente determinada por la configuración anterior en $t - dt$.

Ahora bien, es posible generalizar este "paso" gravitacional del Algoritmo 1 como paso de Markov para cualquier otro tipo de fuerza simulada que dependa de la posición, masa, carga, *flavour*, *spin*, *color* o cualquier otra propiedad relacionada con la partícula. No obstante, se debe remarcar que dicha fuerza simulada no puede depender de propiedades que requieran guardar información de estados previos del sistema dado que el cálculo del nuevo paso no se debe hacer usando información previa al estado actual en el caso que nos atañe.

Entonces, para cualquier fuerza f compatible con la propiedad de Markov, definimos f_{ij} como la magnitud de la fuerza que p_j ejerce sobre p_i. Dicho esto, el paso de Markov $\Delta s_i(t)$ para la partícula p_i en el momento t debería ser;

$$\Delta s_i(t) = \frac{1}{2}\frac{1}{m_i(t)} \sum_{j \neq i} f_{ij}(s_i(t), s_j(t), ...)\vec{n}_{ji}(t)dt(t)^2 \quad (2)$$

donde por definición

$$\Delta s_i(t) \equiv s_i(t + dt) - s_i(t)$$

$$\vec{n}_{ji}(t) \equiv \frac{s_i(t) - s_j(t)}{|s_i(t) - s_j(t)|}$$

tal que, para todas las fuerzas que tengan un factor de dependencia radial, es decir,

$$r_{ij}(t) \equiv |s_j(t) - s_i(t)|\,,$$

finalmente resulta

$$\Delta s_i(t) = \frac{1}{2}\frac{1}{m_i(t)} \sum_{j \neq i} f_{ij}(r_{ji}(t), ...)\vec{n}_{ji}(t)dt(t)^2. \quad (3)$$

B. Fuerza de expansión cosmológica para clustering

Generalizado el paso de Markov en la Eq. 2 supondremos que las únicas propiedades que afectan a la magnitud de la fuerza f en nuestra constribución son las masas m_i y m_j, las distancias entre partículas r_{ij} y la cantidad de clusters c que quedan en el sistema en el momento t.

Estas propiedades construyen la fuerza que, al ser atractiva inicialmente, comienza a ser repulsiva con el paso del tiempo. El objetivo es fusionar inicialmente los clusters, pero, cuando estos hayan crecido lo suficiente, se pretende hacer que sientan una fuerza repulsiva entre sí de tal manera que se favorezca una configuración de equilibrio donde los clusters formados nunca se encuentren.

Para alcanzar este propósito, la fuerza inicialmente debe ser la fuerza gravitacional original, pero a menor número de cúmulos es necesario que sea una fuerza más expansiva. Además, el término de expansión propuesto debe depender de las distancias de tal manera que, cuanto mayor sea la distancia entre las partículas del sistema, más fuerte será la repulsión entre estas. Entonces se propone;

$$f_{ij} = -G\frac{m_i m_j}{r_{ij}^2} + k_c r_{ij}, \quad (4)$$

donde,

$$k_c = \Lambda\left(1 - \frac{c(t)}{n}\right) m_i(t) m_j(t), \quad (5)$$

y $c(t)$ es el número de categorías que quedan en el espacio de simulación en el momento t y n es el número (inicial) de clusters (tantos como datos al inicio), es decir, $c(0) = n$.

Haciendo algunas simplificaciones, el paso de Markov se puede definir como:

$$\Delta s_i(t) = \frac{1}{2}\sum_{j \neq i} m_j(t)\left[-\frac{G}{r_{ij}^2} + \Lambda\left(1 - \frac{c(t)}{n}\right) r_{ij}\right]\vec{n}_{ji}dt(t)^2 \quad (6)$$

C. Reformulación del algoritmo: el parámetro δ

En primer lugar hay que decir que en nuestro enfoque se hace una reformulación del parámetro δ que puede ser aplicada en el original. Es decir, esta reformulación no es una reformulación *ad hoc* que necesita el enfoque cosmológico, todo lo contrario, la reformulación de este parámetro también se puede hacer en la versión original y en cualquier enfoque de algoritmo gravitacional. En resumen, es una modificación para mejorar este tipo de algoritmos generalizados en la Subsección III-A.

La reformulación es la siguiente. El paso de la distancia máxima δ se propone no ser un parámetro constante como lo es en la versión original. En nuestro enfoque, δ será la mitad de la distancia entre la partícula más rápida y su vecino más cercano. En base a este criterio, nos aseguramos de que cuando la fuerza es muy atractiva, la partícula más rápida no pueda intercambiar su posición con su vecina más cercana (ver Fig. 1). Entonces con esta reformulación podemos decir que:

(P1) Si dos partículas están muy cerca una de la otra, y la fuerza que sienten muy atractiva, el paso no puede ser demasiado grande para separarlas y por ende, se ven obligadas a fusionarse.

No obstante, en nuestro enfoque también hay fuerzas repulsivas así que expliquemos qué sucede cuando aplicamos el criterio anterior en otras situaciones.

Cuando comienzan a aparecer fuerzas repulsivas es debido a que la relación $c(t)/n$ es pequeña (ver Eq. ref eq: cosmomark), entonces, se tienen puntos masivos separados cada uno de los otros a grandes distancias y probablemente rodeados de partículas ligeras. En este caso, las partículas

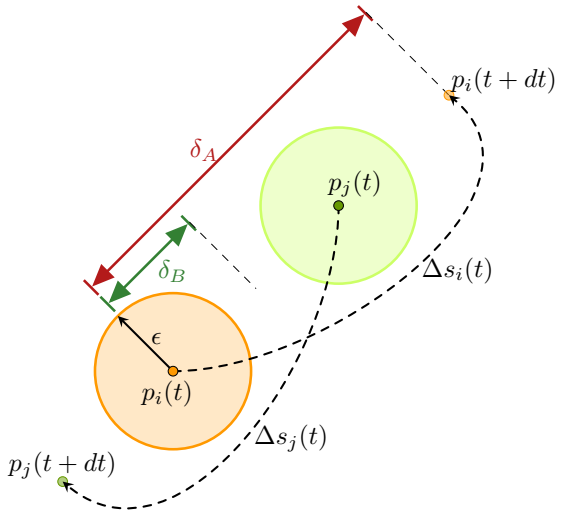


Fig. 1: En esta figura $p_i(t)$ y $p_j(t)$ son la partícula más rápida y su vecino más cercano en un tiempo arbitrario t sintiendo una gran fuerza atractiva entre sí. Si δ es constante y demasiado grande (caso δ_A), las partículas no se pueden fusionar correctamente y pueden intercambiar sus posiciones durante muchas iteraciones. Si definimos δ en base al criterio explicado (caso δ_B), nos aseguramos de que las dos partículas se fusionen.

circundantes hacen que el valor δ sea pequeño, lógicamente, porque la distancia entre estas partículas circundantes y su correspondiente grupo es menor que la distancia entre los grupos.

Entonces, estas partículas ligeras se moverán a posiciones donde estarán más cerca a sus correspondientes clusters que antes, y la fuerza que sentirán será cada vez más atractiva. Pero, no intercambiarán sus posiciones debido a los argumentos explicados anteriormente. Entonces:

(P2) Si hay partículas masivas rodeadas por otras partículas ligeras, el valor de δ será pequeño y además, en dicha situación, los pasos no pueden ser demasiado grandes para alejar la partícula ligera circundante de la masiva (o categoría) a la que pertenece.

Por lo tanto, dadas (P1) y (P2), los estados del sistema siempre van a converger a la configuración donde solo quedan partículas masivas. En este último caso, la distancia entre las partículas es grande, por lo que δ será grande, lo que hará que el sistema explote.

D. Derivación del tiempo de vida de la configuración para el instante t: el valor Δt_c

Debido al criterio anterior para δ, para cada iteración o lapso de tiempo corresponderá un valor de dt derivado de δ: ambos, dependen del tiempo y deben calcularse en cada iteración.

En el momento t, supongamos la partícula más rápida p_{fast} la i -ésima donde se cumple $p_{fast} = p_i$ donde $i = argmax(\Delta s_1(t), \ldots, \Delta s_i(t), \ldots)$, por lo que, una vez detectada esta partícula, se deben calcular tres valores:

- La distancia al vecino más cercano d_{min}, δ y dt para esa iteración.

Por definición y teniendo en cuenta la Eq. 3:

$$\delta \equiv \frac{d_{min}}{2} = \frac{1}{2}|\vec{a}_{fast}|dt^2$$

donde, a_{fast} es la suma de las contribuciones a la aceleración sobre la partícula más rápida, explícitamente;

$$a_i = \sum_{j \neq i} m_j(t) \left[-\frac{G}{r_{ij}^2} + \Lambda \left(1 - \frac{c(t)}{n} \right) r_{ij} \right] \tag{7}$$

luego, es inmediato deducir dt como

$$dt = \sqrt{\frac{d_{min}}{a_{fast}}}. \tag{8}$$

Este dt es muy fácil de calcular y no tiene costo computacional. Una vez se calcula en cada iteración, debe acumularse hasta que se absorba una partícula o, en otras palabras, hasta que cambie la cantidad de partículas $c(t)$. Entonces, definimos la cantidad de tiempo Δt_c donde el sistema ce mantiene con un número c de categorías(o partículas).

$$\Delta t_c \equiv t_{c-1} - t_c = \sum_{t_c}^{t_{c-1}} dt(t) \tag{9}$$

Como condición, si la acumulación de dt excede un valor arbitrario, el algoritmo se ve obligado a parar reconociendo que el sistema explota:

$$\Delta t_c > N \sqrt{\frac{2\left(\frac{D}{10}\right)^3}{3Gn}}, \tag{10}$$

donde, D es la medida de la diagonal del hipercubo N-dimensional que ocupan los datos al principio.

E. Reformulación del algoritmo: el parámetro ϵ

Por otro lado, hemos redefinido el parámetro ϵ. Dicho parámetro va a representar el volumen 3D de la partícula en lugar de ser un parámetro fijo. Este volumen 3D, al igual que la masa, se va a conservar en todo el sistema a lo largo del tiempo. Entonces, cuanto mayor es la masa de una partícula, mayor debe ser su valor ϵ. La regla de la transferencia de volumen es la siguiente. Si en una particula p_j es absorbida por p_i, debido a la conservación del volumen:

$$\frac{4}{3}\pi\epsilon_{new}^3 = \frac{4}{3}\pi\epsilon_a^3 + \frac{4}{3}\pi\epsilon_b^3$$

se obtiene que

$$\epsilon_i(t+dt) = \sqrt[3]{\epsilon_i^3(t) + \epsilon_j^3(t)}, \tag{11}$$

y fijando el valor ϵ_{ini} inicial como una fracción de la distancia entre una partícula aleatoria y su vecino más cercano:

$$\forall i \in [2, n], \epsilon_i(0) = Rand(d_{min})/N \tag{12}$$

finalmente podemos exponer la estructura del algoritmo que proponemos.

Pero, antes de eso, tenemos que decir que la elección de la regla de conservación de volumen 3D y no el 4D o N-D volumen no es arbitraria. Esto se debe a que si nos hemos inspirado en la ley de Gravitación Universal (GU) y las reglas de GU se basan en un universo 3D, estamos obligados a probar esta dimensionalidad al principio. La exploración de otras reglas de dimensionalidad pertenece a otro trabajo o línea futura.

F. Algoritmo de clustering cosmológico

Algoritmo 2: Clustering Cosmológico

Entrada: Un dataset con n datos de N variable
Salida: Una configuración del sistema
Inicialmente asignamos una masa $m_i = 1$ y $\epsilon_i(0)$ de acuerdo con la Eq. 12 a cada partícula p_i;
El tiempo se inicializa a $t = 0$;
Unificación Previa al Ciclo (UPC): Se comprueba qué partículas deben unificarse debido a la elección de ϵ_i inicial. Si dos partículas i y j (o más) están a una distancia menor que $\epsilon_i(0) + \epsilon_j(0)$, su unificación se realiza de acuerdo con la Eq. 11, otorgando a p_i la masa de p_j y eliminando esta última;
mientras $n > 1$ *AND Eq. 10 is fulfilled* **hacer**

1) En la iteracion correspondiente a t, es decir, el lapso $[t, t + dt]$, para cada p_i se calcula su vector de influencia de movimiento $\Delta s_i(t)$ dado por la Eq. 6, donde G y Λ son constantes positivas.
2) Para cada p_i, su nueva posición será:

$$s_i(t + dt) = s_i(t) + \Delta s_i(t)$$

   Estos dos primeros items componen el paso de Markov.
3) Se eleva t a $t + dt$.
4) Si dos partículas i y j están a una distancia menor que $epsilon_i(t) + epsilon_j(t)$, su unificación se realiza de la misma manera que en la UPC.
5) Se categorizan las partículas unídas bajo la misma etiqueta para construir el dendrograma.
6) Se reasigna n.

Una vez se detiene el bucle, evaluamos el historial del sistema (dendrograma) para encontrar la configuración más estable de clústeres. La configuración elegida es la que abarque mayor Δt_c.

Como en el algoritmo original, cada partícula del sistema corresponde a una categoría, no obstante, la duración de todo el proceso depende de si el sistema converge o explota.

Para concluir esta seción, en la Fig. 2 se muestra un esquema del algoritmo propuesto.

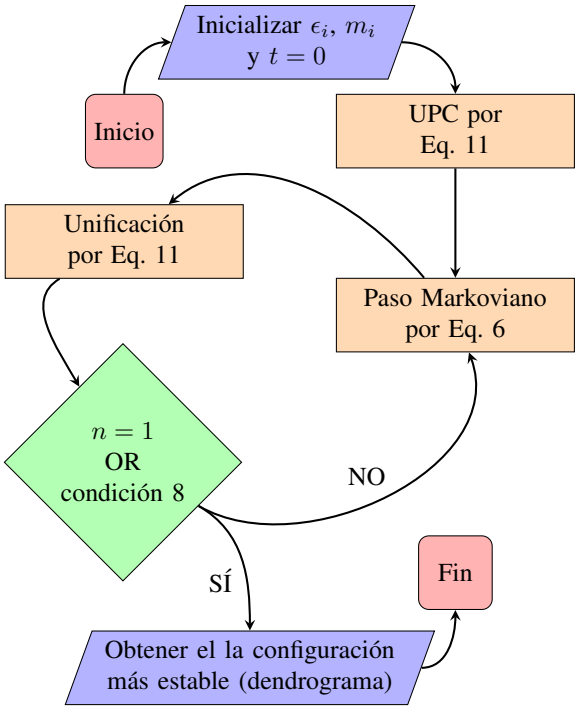
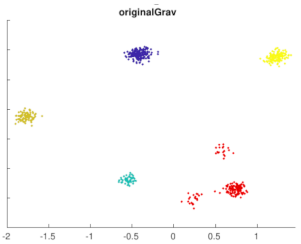
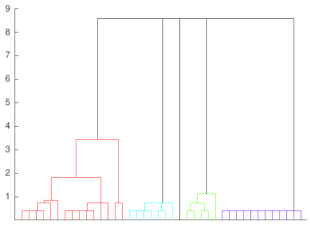


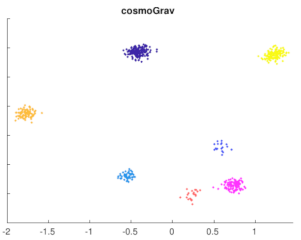
Fig. 2: Esquema del Alg. 2.



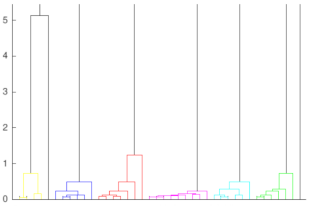
(a) Etiquetado final de clusters generados de manera aleatoria por el algoritmo gravitacional clásico de Wright.



(b) Dendrograma generado por el Alg. 1 para el dataset inicialmente compuesto por seis clusters de la izquierda.

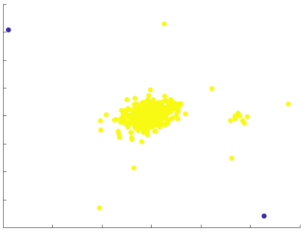


(c) Etiquetado final de los mismos clusters de Fig. 3a pero por el algoritmo cosmológico propuesto en este trabajo.

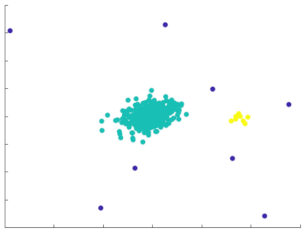


(d) Dendrograma generado por el Alg. 2 para el dataset inicialmente compuesto por seis clusters de la izquierda.

Fig. 3: Diferencias de etiquetado (izquierda) y dendrogramas (derecha) entre los Algoritmos 1 (arriba) y 2 (abajo).

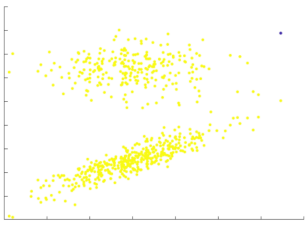


(a) El algoritmo de Wright no consigue diferenciar los clusters ni los outliers.

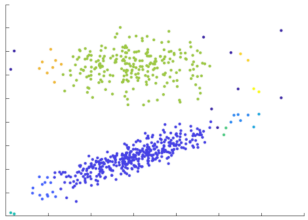


(b) Nuestro algoritmo es capaz de detectar tanto los clusters como los outliers.

Fig. 4: Comparación entre los Algoritmos 1 y 2 para un dataset bidimensional con outliers.



(a) Etiquetado de los datos según el algoritmo gravitacional clásico de Wright.



(b) Etiquetado de los datos según el algoritmo cosmológico propuesto en este trabajo.

Fig. 5: Dataset bidimensional compuesto por dos clusters gaussianos próximos entre sí.

IV. Ejemplos Ilustrativos

En esta sección se recogen tres ejemplos ilustrativos de clasificación en datasets bidimensionales para los diferentes algoritmos.

El primer conjunto de datos sobre el que se han testado los algoritmos es el grupo de clusters distinguibles que aparece en las Figuras 3a y 3c. Acompañando a estas imágenes se muestran también los dendrogramas generados por ambas simulaciones (Figuras 3b y 3d). En este claro ejemplo de la Fig. 3, la configuración más estable en el tiempo de nuestro algoritmo permite clasificar de diferente manera los clusters que se encuentran más cerca entre sí, mientras que son indistinguibles para el algoritmo original (en rojo en la Fig. 3a).

Por otro lado, se ha construido un segundo dataset para la comparación de los algoritmos en entornos con ruido. Posteriormente, se han puesto ambos algoritmos a prueba en el pequeño dataset de la Fig. 4. Es notorio como mientras nuestro algoritmo es capaz de identificar la mayoría de partículas de ruido (Fig. 4b), el original no puede (Fig. 4a).

Por último, se ha construido un tercer dataset que pudiera ser más probable encontrar en problemas de clasificación reales. Este está compuesto por dos clusters de distribución gaussiana que contienen puntos que bien podrian pertenecer a ambas categorías debido a su proximidad (Fig. 5). Como se puede apreciar en la Fig. 5a, el Alg. 1 no ha sido capaz de diferenciar entre ambas clases. Esto es debido a la continua absorción de partículas individuales por parte de los clusters más grandes. Este hecho hace que las configuraciones que pudieran parecer las más estables (más largas en el tiempo simulado), en realidad son truncadas constantemente por la integracion de nuevas partículas a la clase, y finalmente, ambas clases terminan por juntarse. Por otro lado, el Alg. 2 es capaz de separar dos clases *grosso modo* debido a la fuerza repulsiva que perciven las partículas masivas entre sí. Como resultado final se puede apreciar que los extremos de los clusteres son catalogados con diferentes etiquetas creando así pequeños sub-clusters.

V. Conclusiones y líneas futuras

Las mejoras más notorias de nuestra propuesta son;

1) Poder catalogar clusters próximos
2) Que estén inmersos entornos con ruido.

No obstante, poder aunar como una única clase clusters que tengan formas muy alargadas, ya estén bien diferenciados del resto, es un reto que no se consigue completar. Este problema se debe a la construcción del Alg. 2 *per se* ya que se basa en fuerzas radiales. Es por ello que se deberá investigar la implementación de fuerzas no isótropas o de diferentes métricas que distorsionen el espacio.

Finalmente decir que el algoritmo se ha probado en datasets bidimensionales por lo que queda pendiente analizar cómo este se adapta a datasets de mayor dimensionalidad y evaluarlo en un benchmark junto con otros algoritmos de clustering.

Agradecimientos

Este trabajo ha sido respaldado por los proyectos PID2019-108392GB-I00 (AEI/10.13039/ 501100011033) de la Agencia Estatal de Investigación.

References

[1] Armentia, J., Rodríguez, I., Idocin, J. F., Bustince, H., Minárová, M., & Jurio, A. (2019, July). Gravitational clustering algorithm generalization by using an aggregation of masses in newton law. In International Summer School on Aggregation Operators (pp. 172-182). Springer, Cham.

[2] "Kosmologische Betrachtungen zur allgemeinen Relativitätstheorie" ("Cosmological considerations in the General Theory of Relativity"). Consultado el 5 de mayo de 2021.

[3] Blekas, K., & Lagaris, I. E. (2007). Newtonian clustering: An approach based on molecular dynamics and global optimization. Pattern Recognition, 40(6), 1734-1744.

[4] Fumanal-Idocin, J., Alonso-Betanzos, A., Cordón, O., Bustince, H., & Minárová, M. (2020). Community detection and social network analysis based on the Italian wars of the 15th century. Future Generation Computer Systems, 113, 25-40.

[5] Geyer, C. J. (1992). Practical markov chain monte carlo. Statistical science, 473-483.

[6] Tyson, N. D., Strauss, M., & Gott, J. R. (2016). Welcome to the universe: an astrophysical tour. Princeton University Press. Chapter 20

[7] Rugh, S. E., & Zinkernagel, H. (2002). The quantum vacuum and the cosmological constant problem. Studies in History and Philosophy of Science Part B: Studies in History and Philosophy of Modern Physics, 33(4), 663-705.

[8] Wright, W. E. (1977). Gravitational clustering. Pattern recognition, 9(3), 151-166.

Cartografía de un Subcampo Científico: Un Estudio sobre la Optimización de Carteras de Inversión

Feijoo Colomine Durán
Universidad Nacional Experimental del Táchira (UNET),
Laboratorio de Computación de Alto Rendimiento (LCAR),
San Cristóbal, Venezuela
Email: fcolomin@unet.edu.ve

Carlos Cotta*† and Antonio J. Fernández-Leiva*†
*ITIS Software, Universidad de Málaga
†Dept. Lenguajes y Ciencias de la Computación,
ETSI Informática, Universidad de Málaga,
Campus de Teatinos, 29071 - Málaga, España.
Email: {ccottap,afdez}@lcc.uma.es

***Resumen*—En este trabajo de investigación esbozamos una propuesta metodológica para hacer un análisis de un corpus bibliográfico sobre una temática específica. El objetivo fundamental es cartografiar la literatura existente sobre dicha temática, de manera que un usuario interesado en la misma pueda disponer de una cierta estructura a la hora de realizar un estudio de la misma. Partiendo de la información que una consulta bibliográfica en una base de datos de referencia pueda disponer, se considera un análisis para establecer un marco metodológico, centrado en los ítems bibliográficos como elementos de interés. En dicho análisis se intenta realizar una clasificación de los ítems de la literatura en conjuntos relacionados mediante algoritmos de agrupamiento (trabajando en este caso sobre una matriz de distancia entre artículos obtenida a partir de la similitud léxica entre sus descriptores bibliográficos). En este estudio además, se analiza la estructura interna de estos grupos y se intenta encontrar un orden de revisión adecuado mediante el empleo de algoritmos evolutivos. Posteriormente, se realiza un análisis de redes sobre los datos de co-ocurrencia de términos, con el objeto de determinar términos relevantes y asociaciones de interés entre los mismos. Esta metodología se ilustra en este trabajo con un caso de estudio sobre la aplicación de metaheurísticas y algoritmos bioinspirados a problemas de optimización de carteras de inversión.**

***Index Terms*—Algoritmos Evolutivos, Clustering, Análisis de Redes, Análisis Bibliográfico.**

I. Introducción

La capacidad de revisar e informar sobre literatura relevante es una habilidad académica clave. Una revisión de la literatura, puede:

1. Situar el enfoque de investigación dentro del contexto de la comunidad académica más amplia en su campo;
2. Generar un criterio acerca de la literatura que puede parecer relevante;
3. Identificar una brecha dentro de esa literatura que la investigación intentará abordar.

Hasta cierto punto, la revisión de la literatura puede convertirse en un proyecto en sí mismo. Es una muestra importante de los talentos inherentes a la comprensión, interpretación, análisis, claridad de pensamiento, síntesis y desarrollo de argumentos.

El proceso de realizar y reportar una revisión de literatura puede ayudar a aclarar los pensamientos sobre su estudio específico. También puede establecer un marco dentro del cual presentar y analizar los resultados.

La revisión del estado del arte en un tema específico supone hoy en día para los investigadores un esfuerzo muy grande, en un desgaste de recursos y de tiempo. Dicha exploración es importante para comprender los avances que se han dado sobre un problema, las posibles soluciones y en un esquema colaborativo lograr la no repetición de investigaciones para avanzar mucho más rápido.

Uno de los problemas más comunes en la revisión de la literatura resulta en que no se puede discriminar entre los materiales disponibles en las bibliotecas digitales aquellos que pueden ser relevantes a una investigación. Esto puede pasar por el número muy alto de publicaciones que puede tener un tema específico. La clasificación automatizada de la sintáctica a través de los títulos, resúmenes o abstract y palabras claves pueden ayudar a no pasar por alto este problema.

La adopción de los viejos modelos para hacer revisión bibliográfica de un tema específico supone contextualizar y estar en una constante actualización, para poder brindar al investigador de unas bases que le permitan establecer un claro diferenciador de sus nuevos resultados producto de su propia investigación. La admisión de una perspectiva focalizada en su propio nicho de investigación también trae consigo un problema por cuanto no da una visión más amplia del tratamiento que se le da a una investigación en diferentes países.

La clasificación de la relevancia o no de un texto en particular también resulta en un problema al hacer una revisión de un tema específico. El usar herramientas que automáticamente definan relaciones y sistematicen la visión del campo de trabajo en un área determinada ya de por si implica hacer un valioso aporte en el área de investigación que se piense abordar. Nuestro estudio abarca una propuesta metodológica para ordenar la búsqueda de información dado un tema especifico haciendo un desglose y clasificación jerarquizado y permitiendo comprender mejor en que situación real esta alguna temática sujeta de estudio.

II. Materiales y Métodos

En la presente investigación se desarrolla una propuesta metodológica para la selección, análisis, desarrollo y contextualización de contenidos bibliográficos sobre un mecanismo que se encarga de una clasificación de los artículos por eje temático. Además de esto, hemos incorporado un estudio de

la red de palabras que se produce al combinar los términos presentes en todos los artículos

II-A. Propuesta metodológica

Proponemos el desarrollo de un marco metodológico reproducible y que puede ser capaz de aportar una guía para obtener un mecanismo claro y camino para desarrollar un buen análisis documental. Hemos desarrollado un estudio comparativo y análisis de artículos afines a un temario especifico, logrando establecer su grado de correlación y diferenciación. Aunado a esto hemos completado un estudio de la conformación de una red de tópicos que se están abordando en un área de interés especifica (para mayor detalle sobre este tipo de enfoques se sugiere revisar [1]).

Para el marco metodológico, Partimos de un conjunto de datos crudos $\mathcal{A} = \{A_1, \ldots, A_m\}$, donde cada A_i es un ítem bibliográfico descrito mediante una cadena de texto (su título y/o abstract, palabras claves, etc.). Este conjunto de datos se obtiene de una consulta sobre una base de datos bibliográfica relevante (véase la Sección II-B) y ha de ser preprocesado antes de proceder al análisis. Concretamente, realizamos un proceso de lematización (*stemming*)[1] y filtrado que tiene por objetivo (1) eliminar de los descriptores aquellos términos que no aportan nada (por tratarse de palabras conectoras o desprovistas de información semántica relevante para los propósitos del análisis) y (2) eliminar cualquier tipo de flexión gramatical (identificando de esta forma la raíz de cada término con independencia de su morfología concreta, e.g., singular/plural, formas verbales, sufijos, etc.). El resultado es que cada ítem bibliográfico A_i pasa ahora a estar definido por una colección $L_i = \{l_{i,1}, \ldots, l_{i,n_i}\}$ de lexemas.

La dinámica de la primera fase (en un sentido puramente enunciativo, ya que ambas son independientes y podrían realizarse en cualquier orden) es ahora como sigue. En primer lugar, realizamos una partición de los ítems bibliográficos en conjuntos relacionados. Esto puede realizarse de diferentes formas dependiendo del método de agrupamiento que se considere. En nuestro caso, una solución efectiva puede ser el empleo de un algoritmo clásico como el de las k-medias [2], [3], [4]. Este algoritmo recibe una matriz M de tamaño $m \times p$ donde cada fila (una por ítem bibliográfico) es un vector p-dimensional (en nuestro caso estos son los vectores de incidencia de la colección de términos del ítem correspondiente sobre el conjunto total de términos; esto es, p es la cardinalidad del conjunto $\cup_i L_i$, y $M_{i,j} = 1$ si L_i contiene el término j-ésimo y $M_{i,j} = 0$ en otro caso), y trata de agrupar iterativamente los elementos en k grupos, determinando en cada paso el centroide del mismo y asignando cada elemento en el siguiente paso al grupo cuyo centroide es más cercano. En este trabajo la distancia a los centroides (la media de los vectores de los elementos del grupo) se determina mediante el complemento de la similitud coseno [5], y el número de grupos se determina a posteriori mediante el método Silhouette [6].

[1]En este trabajo hemos empleado la biblioteca de lematización disponible en el *Text Analytics Toolbox*™ de Matlab®.

Una vez disponemos de los grupos, se procede a determinar un orden de los ítems del mismo. Dicho orden tiene por objeto proporcionar al usuario una guía mediante la cual revisar secuencialmente el grupo de tal forma que ítems en posiciones adyacentes sean lo más similares posibles (y por lo tanto se facilite el análisis de estos elementos bibliográficos). Para ello, se necesita calcular una matriz de distancias $D = \{d_i, j\}_{i,j=1\ldots m}$ entre artículos. Dicha matriz puede calcularse de numerosas formas. En este trabajo hemos optado por simplicidad por una medida sencilla basada en la proporción de términos comunes entre artículos. De manera más precisa, consideramos

$$d_{i,j} = 1 - \frac{|L_i \cap L_j|}{|L_i \cup L_j|}$$

donde L_i, y L_j son las colecciones de lexemas asociadas a los ítems bibliográficos concretos, tal como se indicaba anteriormente. Así, la distancia queda definida como el complemento del grado de similitud de conjuntos de Jaccard [7]. El problema al que nos enfrentamos es por lo tanto el de encontrar un camino hamiltoniano de peso mínimo sobre la base de esta matriz de distancia (obviamente considerando en cada caso únicamente los ítems incluidos en cada grupo). Este es un problema NP-duro para el que pueden emplearse diferentes heurísticas. En este trabajo hemos empleado un método reminiscente de enfoques desarrollados para análisis filogenético [8]. Concretamente, determinamos una clasificación arbórea de los elementos del grupo y –explotando intercambios izquierda-derecha dentro del árbol– buscamos una disposición de las hojas de manera que un recorrido de izquierda a derecha minimize la suma de distancias entre elementos adyacentes. El cálculo de la clasificación en árbol se realiza mediante un algoritmo de agrupamiento aglomerativo mediante *average-linkage*, y la determinación de los intercambios izquierda-derecha que minimizan la distancia resultante mediante un algoritmo evolutivo[2].

En cuanto al análisis de términos se parte del conjunto $\mathcal{L} = \{L_1, \ldots, L_m\}$ descrito anteriormente siguiendo en parte lo presentado en [1]. A partir de este conjunto se considera la colección global de términos usados $U = \cup_{i=1}^{m} L_i$, de donde eliminamos aquellos términos que solo aparecen una sola vez para quedarnos sólo con términos significativos a nivel de la muestra bibliográfica[3]. Hecho esto, se construye un grafo con pesos $G(V, E, W)$ en el que

- V es el conjunto de términos seleccionados.
- $E = \{(u, v) \mid \exists i : (u \in L_i) \wedge (v \in L_i)\}$, esto es, una arista entre términos que aparecen en un mismo ítem.
- $W((u, v)) = \sum_{i=1}^{m} [\{u, v\} \subseteq L_i]$, donde $[\cdot]$ son los paréntesis de Iverson. Esto es, el peso de cada arista es el

[2]Nótese que en un árbol binario con n hojas hay $n - 1$ nodos internos, en cada uno de los cuales se puede aplicar un intercambio izquierda/derecha. Por lo tanto, dentro del algoritmo evolutivo las soluciones son vectores binarios de longitud $n - 1$, pudiendo emplearse operadores estándar para recombinación y mutación.

[3]De hecho, sería posible fijar un umbral de relevancia más alto. Para esto podemos tener en cuenta de alguna manera la amplitud de la muestra bibliográfica considerada. Este es un aspecto que no se ha considerado en este trabajo.

número de veces que dos términos aparecen en el mismo item.

Una vez determinado este grafo se aplican algoritmos de detección de comunidades para identificar subconjuntos de términos frecuentemente emparejados. Esto subconjuntos representan subdominios temáticos dentro de la muestra bibliográfica considerada, cuya consistencia con los conjuntos anteriormente identificados vamos a estudiar. En este trabajo hemos considerado el empleo del algoritmo de modularidad de Blondel et al. [9] para esta fase.

II-B. Datos usados en el análisis

Para realizar un caso de estudio de la metodología anterior hemos escogido DBLP[4] como fuente de datos. Esta base de datos, mantenida por el equipo de DBLP en la Universidad de Trier, proporciona información bibliográfica sobre las principales revistas y actas de conferencias de informática, enumerando más de 3 millones de publicaciones. Constituye por lo tanto un recurso muy comúnmente empleado para realizar búsquedas bibliográficas en el área que nos ocupa. En este trabajo hemos elegido como caso de estudio la aplicación de técnicas metaheurísticas a la optimización de carteras de inversión. Para ello, hemos realizado una búsqueda sobre DBLP para recuperar artículos que contienen al menos un término relacionado con las metaheurísticas y al menos un término relacionado con las carteras de inversión. Con respecto a lo primero, los términos considerados son:

- EVOLUTIONARY
- GENETIC
- SWARM
- ACO
- METAHEURISTIC
- TABU

En cuanto a lo segundo, hemos considerado:

- PORTFOLIO
- INVESTMENT
- MARKOWITZ
- SHARPE

Obviamente, tanto en un caso como en otro podrían haberse considerado términos adicionales, pero la muestra bibliográfica obtenida es en cualquier caso lo suficientemente amplia como para realizar un primer estudio. Concretamente, la consulta anterior (realizada en mayo de 2021) devolvió 374 artículos. Nótese que cada uno de estos artículos está caracterizado únicamente por su título (otras bases de datos bibliográficas pueden proporcionar además el resumen de los artículos, pero no es este es caso).

III. Experimentación

En esta sección vamos a mostrar todo el proceso de experimentación desarrollado según el marco definido en la sección II.

[4]https://dblp.uni-trier.de/

III-A. Análisis comparativos de artículos afines a un temario investigador

Como se describe en la sección II-B la consulta a la base de datos arroja 374 artículos que tienen como base temática la aplicación de diferentes técnicas metaheurísticas a la optimización de carteras de inversión. La Figura 1 (a) muestra una nube de palabras que permite apreciar la frecuencia absoluta de los términos empleados en los títulos de los trabajos (recuérdese que estos términos está lematizados). Lógicamente, los términos directamente asociados a las palabras clave de búsqueda sobresalen claramente, pero puede apreciarse también la presencia de otros términos relacionados que nos permiten determinar temas y conceptos relacionados que serán de interés (por ejemplo, *investment*, *multiobjective* o *fuzziness*, por mencionar algunos ejemplos). Al margen de proporcionar información valiosa sobre los aspectos que pueden ser de interés en relación con la temática considerada, estos términos puede ser también importantes a la hora de estructurar la bibliografía. Para ver esto, procedemos al siguiente paso anteriormente mencionado, esto es, la partición del conjunto de datos en subgrupos (empleando el algoritmo de las k-medias tal como se indicó). En este caso, de acuerdo con la métrica Silhouette determinamos que el número óptimo de grupos es de 5. La Figura 1 (b)–(f) muestra una primera perspectiva de estos grupos y los términos que emplean. Como puede apreciarse, en general estos grupos se organizan metodológicamente atendiendo a la herramienta de optimización considerada preferentemente.

Para obtener una visión más fina y una mejor caracterización de estos grupos, la Tabla I muestra el tamaño y los términos más comunes en estos grupos. Para obtener información más clara acerca de los términos que resultan más distintivos en cada uno de los grupos, asociamos a cada término un peso que corresponde a su frecuencia absoluta ponderada por cuánto más frecuentemente se usa ese término en el grupo en cuestión frente al conjunto global. De manera más precisa, sean $G_1, \ldots, G_k$ los k grupos identificados. Ahora, sea $c_{i,j}$ definido como

$$c_{i,j} = \sum_{s \in G_i} M_{s,j}$$

esto es, la frecuencia absoluta de utilización del término j-ésimo en el grupo i-ésimo Si definimos la frecuencia global c_j como

$$c_j = \sum_{i=1}^{k} c_{i,j}$$

entonces podemos definir las frecuencias relativas $f_{i,j}$ y f_j como:

$$f_{i,j} = \frac{c_{i,j}}{\sum_{r=1}^{p} c_{i,r}}$$

$$f_j = \frac{c_j}{\sum_{r=1}^{p} c_r}$$

y el peso $w_{i,j}$ de cada término en un grupo como

$$w_{i,j} = c_{i,j} \cdot \frac{f_{i,j}}{f_j}$$

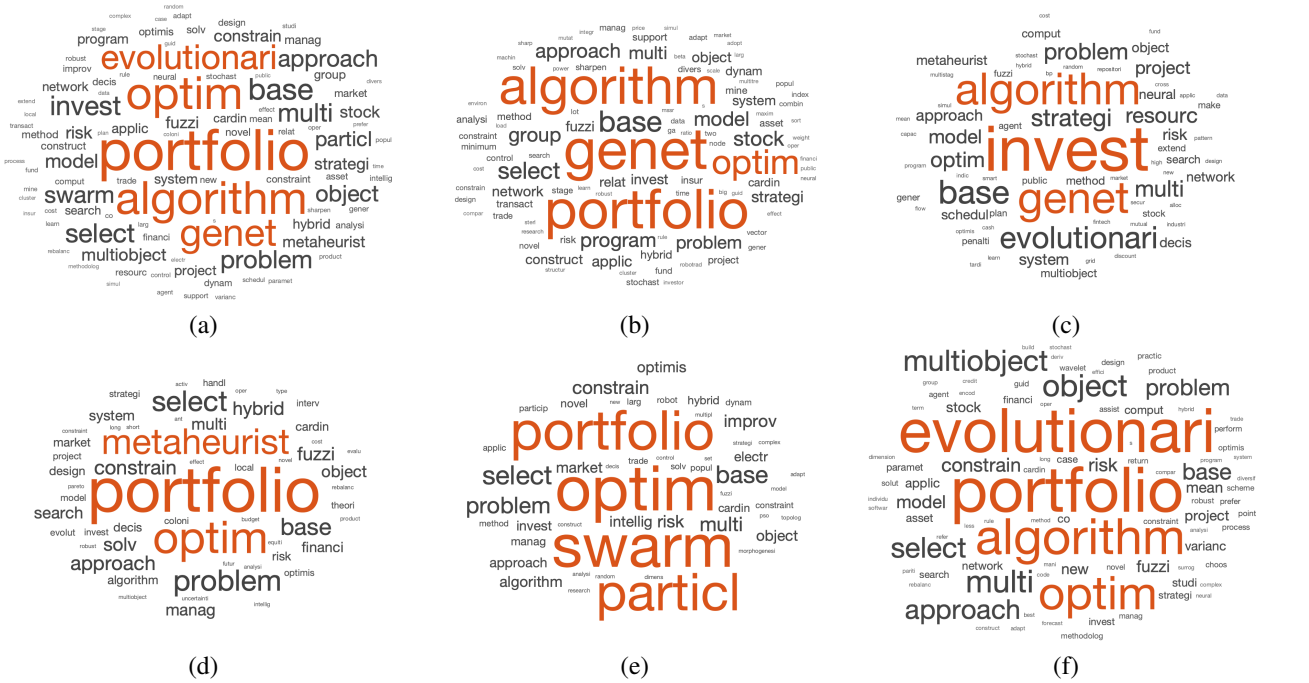


Figura 1. La imagen (a) muestra la nube de palabras representativa de la frecuencia absoluta de todos los términos en el conjunto total de artículos. Las restantes imágenes (b)–(f) muestran lo mismo para cada uno de los cinco grupos identificados (de izquierda a derecha y de arriba a abajo, grupos 1, 2, 3, 4, y 5).

Tabla I
GRUPOS DE ARTÍCULOS IDENTIFICADOS Y TÉRMINOS MÁS DISTINTIVOS DE CADA UNO DE ELLOS.

	Grupo 1		Grupo 2		Grupo 3		Grupo 4		Grupo 5	
Tamaño	113	(30.5 %)	55	(14.7 %)	49	(13.1 %)	56	(15.0 %)	100	(26.7 %)
Lema (peso)	genet	698,27	invest	419,08	metaheurist	162,59	swarm	14779,77	evolutionari	1120,95
	algorithm	148,79	resourc	85,37	portfolio	38,17	particl	4403,45	multiobject	137,24
	group	144,66	schedul	26,35	evolut	36,48	optim	121,78	portfolio	105,12
	portfolio	96,20	genet	23,12	solv	24,83	electr	113,73	varianc	77,20
	program	72,33	strategi	18,22	optim	18,47	improv	44,58	algorithm	66,62

Si observamos los 5 términos con más peso dentro de cada uno de los grupos, podemos apreciar que los grupos más grandes (el grupo 1 y el grupo 5) gravitan en torno a programación genética y en torno a algoritmos evolutivos multiobjetivo respectivamente. Es interesante esto último ya que la búsqueda bibliográfica no incluía directamente términos relativos a la optimización multiobjetivo, pero el análisis automatizado ha sido capaz de identificar esta importante vertiente en relación con la selección de carteras de inversión. Los otros tres grupos son de tamaño algo menor y también se organizan en torno a diferentes técnicas. Así, en el grupo 4 resalta sobremanera la optimización mediante enjambres de partículas. Por su parte en el grupo 3 resalta el empleo del término *metaheuristic*. De hecho, en este grupo también se incluyen otros términos como *ant* entre los 10 más utilizados (peso = 16,21). El grupo 2 parece tener un componente importante de algoritmos genéticos, pero no es el término más empleado ni más distintivo. Cabe de hecho reseñar que en este grupo aparece el término *neural* en el puesto 10 (peso = 13,17), lo que hace atisbar la mayor heterogeneidad metodológica de este grupo de trabajos. En este sentido, parece que este grupo está más orientado hacia aspectos más financieros que algorítmicos. Así, resaltan los términos *investment*, *resource* y *scheduling*, relativos a la planificación de inversiones y a tipos de estas. El término *mutual*, correspondiente a un tipo de fondos de inversión también es uno de los que se pueden encontrar en este grupo (puesto 19, peso = 12,65).

El siguiente paso es realizar una ordenación de los trabajos incluidos en cada grupo con objeto de proporcionar al investigador una guía de lectura/revisión de los mismos. Para ello se ha empleado una algoritmo evolutivo operando sobre un agrupamiento jerárquico de estos trabajos, tal como se describió en la Sección II-A. En este trabajo hemos considerado un algoritmo evolutivo generacional con elitismo, con una población de 50 individuos, un número máximo de evaluaciones de 25000, cruce de un punto ($p_X = 0{,}9$) y mutación binaria bit-flip ($p_m = 1/\ell$, donde ℓ es la longitud del genoma; en este caso $\ell = |G_i| - 1$, como se apuntó anteriormente). La Tabla II muestra un extracto de esta ordenación para uno de los grupos considerados. Nótese como hay la secuencia

Tabla II
EXTRACTO DE LA ORDENACIÓN DE ARTÍCULOS CORRESPONDIENTE AL GRUPO 2.

Data Transformation Methods for Genetic-Algorithm-Based Investment Decisions.
A Decision Investment Model to Design Manufacturing Systems based on a genetic algorithm and Monte-Carlo simulation.
A Sustainable Energy Investment Planning Model Based on the Micro-Grid Concept Using Recent Metaheuristic Optimization Algorithms.
Maximisation of investment profits - An approach to MACD based on genetic algorithms and fuzzy logic.
Extraction of investment strategies based on moving averages – A genetic algorithm approach.
Optimizing investment strategies based on companies earnings using genetic algorithms.
Genetic Algorithm-based Optimal Investment Scheduling for Public Rental Housing Projects in South Korea.
Multi-objective Optimal Public Investment - An Extended Model and Genetic Algorithm-Based Case Study.
A Genetic Algorithm-Based Double-Objective Multi-constraint Optimal Cross-Region Cross-Sector Public Investment Model.
Multi-layered security investment optimization using a simulation embedded within a genetic algorithm.
Combining Support Vector Machine with Genetic Algorithms to optimize investments in Forex markets with high leverage.
Genetic algorithms to optimise the time to make stock market investment.
A novel methodology for stock investment using high utility episode mining and genetic algorithm.
The Random Neural Network with a Genetic Algorithm and Deep Learning Clusters in Fintech – Smart Investment.
Fintech Bitcoin Smart Investment Based on the Random Neural Network with a Genetic Algorithm.
Application of BP Neural Network Based on Genetic Algorithm Optimization in Evaluation of Power Grid Investment Risk.
The Modeling of Genetic and Tabu Search Algorithm Based BP Neural Network in the Risk Analysis of Investment.
Kohonen's neural network and evolutionary algorithms in searching for financial investment strategy.

de trabajos guarda cierta consistencia, lo que puede facilitar la revisión de los mismos al permitir considerar temáticas relacionadas de manera sucesiva. Obviamente, la ordenación se ha realizado mediante un procedimiento heurístico, por lo que no hay garantía de optimalidad y ciertamente puede ser posible encontrar mejores ordenaciones ajustando los parámetros del algoritmo o incluso empleando otro enfoque de optimización. Nótese también un aspecto tangencial a lo anterior pero también relevante: en toda búsqueda bibliográfica van a aparecer con gran probabilidad falsos positivos que –aunque satisfagan los criterios de búsqueda– no se correspondan con la temática. Si hay un factor común en estos falsos positivos, la metodología mostrada los va a identificar y agrupar, lo que facilitará su posterior cribado.

III-B. Red de tópicos

Hemos cumplimentado este estudio agregando un análisis con la red compleja obtenida a partir de la co-ocurrencia de términos en los ítems bibliográficos. Dicha red se muestra en la Figura 2. Hay diferentes tipos de análisis que pueden realizarse sobre esta red (cf. [1]). De entrada, puede aplicarse un algoritmo de detección de comunidades cuya finalidad sería la de identificar subgrupos temáticos de interés, definidos por una especial interacción entre ciertos términos. Un análisis siguiendo el criterio de modularidad de Blondel et al. [9] arroja 7 comunidades (diferentes colores en la Figura 2). No obstante, cuatro de ellas son muy pequeñas (entre 6 y 12 términos). En algún caso corresponden a términos que sólo se usan en un conjunto muy reducido de ítems bibliográficos (por ejemplo, *bee* y *colony*, correspondientes a una técnica de optimización, o *transaction* y *costs*, correspondientes a una especificidad del problema). Las comunidades más grandes son a grandes rasgos consistentes con la partición de los trabajos. Así, hay una comunidad de 107 términos que es consistente con los grupos 1 y 2 anteriores (algoritmos genéticos, programación genética, planificación de inversiones), y otra gran comunidad de 127 términos que se alinea con los grupos 3 y 5 (metaheurísticas y algoritmos evolutivos multiobjetivo). De manera interesante, una pequeña comunidad de 20 términos coincide con la caracterización del grupo 4 relativa a optimización mediante enjambres de partículas.

IV. CONCLUSIONES

En este trabajo hemos presentado un esbozo de una metodología para automatizar el cartigrafiado de un tema de interés a partir de la literatura existente sobre el mismo. El uso de esta técnica metodológica permite identificar las comunidades de autores y de investigación que sobresalen en el área de estudio, además de que facilita determinar cuáles son las áreas de abordaje en la solución de un problema determinado que más frecuentemente se emplean en la literatura. Adicionalmente, la identificación de subtemas permite hacer una clasificación bibliográfica en mejor condición. El hecho de hacer un análisis exhaustivo del conjunto de artículos publicados en un tema especifico, además de establecer un mecanismo de clasificación automatizado permite una simplificación bastante importante del esfuerzo en la revisión del estado del arte.

Para ello, la propuesta presentada establece un análisis de similitud de conjuntos de artículos a través de una métrica de distanciamiento que permite agrupar o segmentar estos a través de diferentes técnicas algorítmicas y que posteriormente permite ordenar secuencialmente estos ítems, lo que nos da una ruta para desarrollar una revisión bibliográfica exploratoria que se supone de mayor relevancia para una investigación documental dada. Si bien se han presentado opciones concretas para llevar a cabo cada una de estas fases del análisis, es evidente que existe la posibilidad de emplear otras herramientas algorítmicas de propósito equivalente. Esto se puede extender del mismo modo al análisis de la red compleja de términos empleados en la literatura seleccionada. En este sentido, una de las líneas de trabajo que pretendemos desarrollar está orientada a la realización de un análisis temporal de esta red de términos en la línea de lo realizado en [11], con objeto de identificar patrones temáticos emergentes.

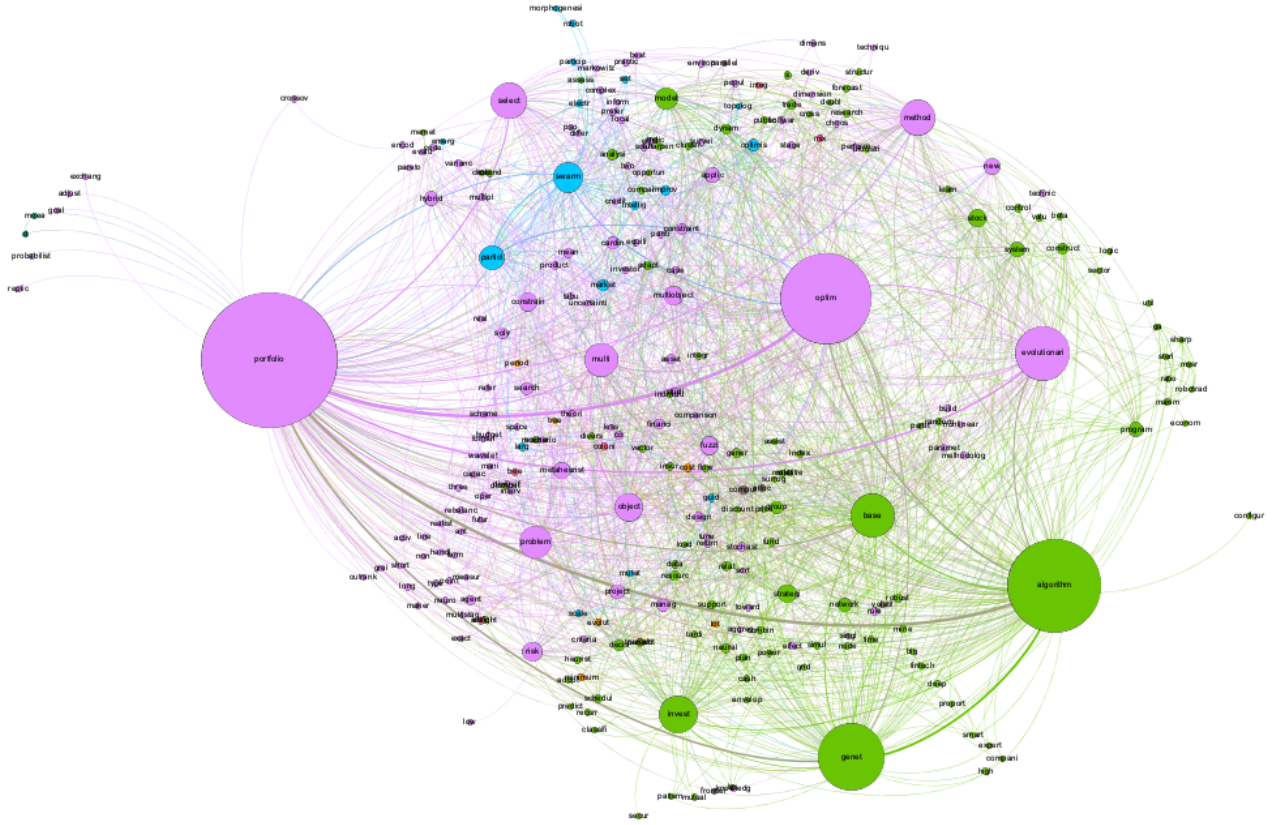


Figura 2. Red de tópicos obtenida a partir de la muestra bibliográfica. El tamaño de cada nodo o arista es proporcional a su frecuencia, y cada color corresponde con una comunidad. La representación gráfica y la determinación de comunidades se ha realizado con Gephi [10].

Agradecimientos

Este trabajo está financiado en parte por el Ministerio Español de Economía y Competitividad (TIN2017-85727-C4-1-P, https://deepbio.wordpress.com/) y por la Universidad de Málaga, Campus de Excelencia Internacional Andalucía Tech.

Referencias

[1] M. Cobo, A. López-Herrera, E. Herrera-Viedma, and F. Herrera, "An approach for detecting, quantifying, and visualizing the evolution of a research field: A practical application to the Fuzzy Sets Theory field," *J. Informetrics*, vol. 5, pp. 146–166, 01 2011.

[2] S. P. Lloyd, "Least squares quantization in PCM," Bell Lab, Tech. Rep. RR-5497, 1957.

[3] J. B. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Fifth Berkeley symposium on mathematical statistics and probability*, L. M. Le Cam and N. J., Eds., vol. 1. California: University of California Press, 1967, pp. 281–297.

[4] J. A. Hartigan and M. A. Wong, "Algorithm AS 136: A K-Means Clustering Algorithm," *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, vol. 28, no. 1, pp. 100–108, 1979.

[5] A. Singhal, "Modern information retrieval: A brief overview," *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering*, vol. 24, no. 4, pp. 35–43, 2001.

[6] P. J. Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," *Computational and Applied Mathematics*, vol. 20, pp. 53–65, 1987.

[7] N. Chung, B. Miasojedow, M. Startek, and A. Gambin, "Jaccard/Tanimoto similarity test and estimation methods for biological presence-absence data," *BMC Bioinformatics*, vol. 20, 12 2019.

[8] P. Moscato, L. Buriol, and C. Cotta, "On the analysis of data derived from mitochondrial DNA: Kolmogorov and a traveling salesman give their opinion," in *Advances in Nature Inspired Computation: the PPSN VII Workshops*, D. Corne *et al.*, Eds. PEDAL, University of Reading, 2002, pp. 37–38.

[9] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2008, no. 10, p. P10008, Oct 2008.

[10] M. Bastian, S. Heymann, and M. Jacomy, "Gephi: An Open Source Software for Exploring and Manipulating Networks," in *International AAAI Conference on Weblogs and Social Media*. AAAI Press, 2009, pp. 361–362.

[11] R. Lara-Cabrera, C. Cotta, and A. Fernández-Leiva, "An analysis of the structure and evolution of the scientific collaboration network of computer intelligence in games," *Physica A: Statistical Mechanics and its Applications*, vol. 395, pp. 523–536, 2014.

A la desinformación le gusta la compañía: Representación de bulos de Twitter sobre la COVID-19 mediante *embeddings*

Guillermo Villar-Rodríguez*, Javier Huertas-Tato*, Alejandro Martín*, David Camacho*
*Departamento de Sistemas Informáticos
Universidad Politécnica de Madrid
Madrid, Spain
Email: {guillermo.villar,javier.huertas.tato,alejandro.martin,david.camacho}@upm.es

***Abstract*—La desinformación sobre la pandemia de la COVID-19 ha tenido un gran impacto en la sociedad. Son numerosos los rumores y bulos extendidos en las redes sociales por individuos que niegan las vacunas o la existencia del coronavirus. En esta investigación, analizamos 20 bulos distintos sobre la COVID-19 expandidos en países de habla hispana. Con este fin, presentamos una metodología para recoger tuits relacionados con bulos en LatamChequea Coronavirus, una base de datos de noticias verificadas, con el objetivo de construir con ellos una representación gráfica a través de *embeddings* semánticos procedentes a partir de un modelo *transformer* pre-entrenado multilingüe: DistilBERT. Además de aplicar reducción de dimensionalidad, nuestra metodología cuenta con una red neuronal profunda para separar más los *embeddings* de bulos específicos. De esta forma, nuestro enfoque pretende explorar los patrones que se originan entre los diferentes tipos de información falsa.**

***Index Terms*—COVID-19, Desinformación, Fact-checking, Verificación, Procesamiento del Lenguaje Natural, Deep Learning, Transformers**

I. Introducción

La irrupción de la pandemia de la COVID-19 ha dado lugar a una infodemia en paralelo debido a la expansión de desinformación, (planeada o no, en alusión a los términos ingleses *disinformation* y *misinformation*, respectivamente), sobre este tema. Ya en febrero de 2020, la Organización Mundial de la Salud advirtió de este aumento de información falsa, y Cristina Tardáguila, Directora Asociada del *International Fact-checking Network*, lo describió como el "mayor reto" para los verificadores [1].

Bajo este contexto sin precedentes, el aprendizaje automático y, en particular, técnicas de *Deep Learning*, pueden ser muy valiosos como herramientas para reducir los diferentes tipos de información falsa sobre el coronavirus. T.T. Nguyen, en su estudio sobre las direcciones que puede tomar la inteligencia artificial para combatir esta pandemia, incluye la lucha contra la información falsa entre los principales enfoques a seguir [2]. Asimismo, en su investigación análoga, Shorten, Khoshgoftaar y Furht subrayan el uso de técnicas de Procesamiento del Lenguaje Natural para para la detección de bulos, entre otras prácticas como el análisis de sentimiento en Twitter como ejemplo de plataforma para realizar esta tarea [3].

Sin embargo, también hay que prestar atención a Twitter como plataforma de diseminación de rumores y bulos, y esto nos invita a desarrollar otro tipo de análisis. Más allá de clasificar la desinformación sobre la pandemia, nuestra investigación busca comprobar cómo interactúan los distintos grupos de temas entre ellos en términos de proximidad e, idealmente, descubrir la naturaleza de esta información falsa en Twitter como punto de partida para combatirla.

Nuestra búsqueda espera ahondar en las relaciones entre las falsedades y las publicaciones *online* que las extienden, lo cual es beneficioso en términos de explicabilidad. Esto es relevante porque mientras áreas como el periodismo demuestran a través de la verificación si una afirmación es verdadera o falsa, la detección de desinformación a través de los algoritmos puede asemejarse a una caja negra [4].

Además, la crecida masiva de los *fact-checks* sobre la COVID-19 de enero a marzo, al comienzo de la pandemia, sugiere dos ideas [1]: por un lado, que los esfuerzos en la investigación relativa a la IA también sirven para descubrir cómo se expande la infodemia, aparte de paliar sus efectos en tareas dedicadas a la clasificación de información; por otro lado, que el número creciente de verificaciones publicadas por organizaciones de *fact-checking* pueden servir también después como recurso para afrontar este problema.

Por esta razón, la intención de nuestra investigación es descubrir cómo los tuits sobre información falsa cuyas verificaciones hayan sido ya documentadas se relacionan entre ellos cuando los transformamos en *embeddings* semánticos. Estos experimentos están solo centrados en el uso de textos de habla hispana, más que tomar una panorámica global.

Este artículo se estructura de la siguiente forma: primero, la Sección 2 introduce el marco teórico detrás del agrupado de informaciones y falsedades sobre la COVID-19; después, la Sección 3 describe los pasos para obtener publicaciones falsas en Twitter, que serán nuestra fuente de datos para nuestro enfoque basado en Análisis de Componentes Principales de *embeddings transformer* combinados con un clasificador neuronal; por último, en la Sección 4 se encuentra la representación gráfica de estos grupos de falsedades, con sus dimensiones reducidas y el conocimiento extraído de ellas,

como antesala a la conclusión en la Sección 5.

II. Marco teórico y trabajos relacionados

En el análisis del discurso en redes sociales, estudiar las representaciones del espacio derivado de los textos que lo componen constituye una tarea relevante en el campo del Procesamiento del Lenguaje Natural. En este aspecto, la extracción de las características (*feature extraction*) de contenido textual ha pasado de métodos que parten del conteo de palabras, como las bolsas de palabras y los *tf-idf*, a representaciones vectoriales complejas de *embeddings* [5].

Un *word embedding* es un vector que representa una palabra en un espacio de *n* dimensiones a partir de sus propiedades. El término se emplea en concreto para las representaciones como las generadas por Glove [6] o word2vec [7], entre otras, que se refieren a vectores densos obtenidos al pasar una red neuronal a un *corpus* para capturar así el significado de la palabra y sus relaciones respecto al resto de palabras en el espacio dimensional [8]. Así, una frase puede recoger su significado en un *embedding* a través de un vector promedio de los vectores de sus palabras y, si dos *embeddings* de publicaciones en redes mediante este procedimiento están representados muy cerca el uno del otro, podemos afirmar que son semánticamente muy similares [5].

En este sentido, los *transformers* suponen un avance dentro de los tipos de *embeddings* semánticos. Mientras que con los tipos de *embeddings* mencionados antes se construye un vector de la palabra sin tener en cuenta que tendrá un significado diverso en función de donde se encuentre, los *embeddings* generados por *transformers* (por ejemplo, ELMo [9] o BERT [10]) solventan este inconveniente por tratarse de vectores distintos dependiendo del contexto en el que la palabra se emplea [5].

La expansión de los *posts* en línea sobre la COVID-19 ha motivado las investigaciones para mapear el discurso sobre el coronavirus en las redes sociales y, por tanto, también la aplicación de estas técnicas mencionadas. La hoja de ruta común para dibujar los temas a los que se refiere la conversación en internet sobre la COVID-19 consiste en *embeddings* semánticos junto a algoritmos y técnicas adicionales para mejorar la representación gráfica final de este proceso, siempre que sea necesario [5], [11], [12]. Sin embargo, si indagamos en los detalles específicos, encontramos diferencias. Por ejemplo, Asgari-Chenaghlu, Nikzad-Khasmakhi y Minaee tomaron este camino en sustitución a métodos más tradicionales como la Asignación Latente de Dirichlet [11], mientras que Chen, Zhong y Pang prefirieron combinar los *embeddings* contextuales de BERT con los vectores de la ALD [12], en vez de prescindir de ellos.

No obstante, aunque Chen et al. utilizaron la palabra "información" para referirse a la representación del espacio latente de los temas surgidos de la COVID-19, los autores encontraron un *cluster* de tuits sobre racismo y su sistema "no detectaba información engañosa publicada por bots" [12], lo cual podría causar que se metan bulos en cada uno de los temas encontrados. Esto significa que estas representaciones visuales de las características textuales de la COVID-19 quizás no separen información de desinformación correctamente.

Teniendo esto en cuenta, se necesitarían nuevos estudios que mapeen las falsedades sobre la COVID-19 extraídas del cúmulo de textos publicados en las redes sociales. De esta forma, se entendería la naturaleza de esta infodemia, más que centrarnos en el hecho de solo descubrir los temas de las publicaciones sobre la COVID-19 sin preocuparnos de si son verdaderas o falsas. Las técnicas explicadas arriba, desde los *tf-idf* a los *embeddings* semánticos que tienen en cuenta el contexto (a través de modelos *transformer*), ya han sido implementadas para detectar información falsa sobre la COVID-19 [13], incluso con modelos *transformer* para *corpora* sobre el coronavirus [14], pero todas estas técnicas también pueden servir como inicio para visualizar el espacio latente y analizar los grupos que se forman entre *embeddings*, como se ha visto antes.

Nuestra investigación pone el foco en esta metodología para la visualización de desinformación. En esta línea, Ng and Carley ya probaron antes la detección de tuits con falsedades en ellos usando *clusters* de historias desmentidas por varias organizaciones de *fact-checking* [15]. Tener los *fact-checks* como fuente primaria para nuestros experimentos nos ayuda a extraer la mayor cantidad posible de tuits sobre las piezas de desinformación detectadas por los *fact-checkers*, dejando aparte los tuits que no están relacionados. Haciendo esto, no tendríamos que recoger tuits aislados de una base de datos de información falsa, sino que recurriríamos a las noticias de *fact-checking* para descargar todos los tuits que puedan encajar con el bulo negado oficialmente por al menos una organización de *fact-checking* en un intervalo específico de tiempo.

A. Metodología

En esta sección, describimos la metodología para extraer y procesar los *embeddings* semánticos y para representar cómo los tuits que expanden la desinformación sobre la COVID-19 interactúan y evolucionan con el tiempo. Nuestro enfoque comienza con la recolección de una lista de bulos ya descubiertos y desmentidos por verificadores. Una vez recogidos, se extraen los tuits que aluden a cada uno de esos bulos como pista para analizar qué información falsa se despliega rápidamente en Twitter y cómo hay bulos que están relacionados entre sí. Tras esto, transformamos tales tuits en los vectores de *embeddings* y reducimos sus dimensiones para que la representación final de cada texto sea más útil. Finalmente, entrenamos un modelo de clasificación para aumentar la separación entre bulos y estudiar las relaciones entre ellos. Este proceso se ilustra en Fig. 1 y se detalla en los siguientes apartados.

B. Extracción de datos

Los tuits usados para nuestro experimento se han obtenido a través de LatamChequea Coronavirus [16], el principal *dataset* de falsedades en español sobre el coronavirus de acuerdo a la IFCN, completado por 35 organizaciones de verificación de habla hispana, coordinadas por Chequeado.

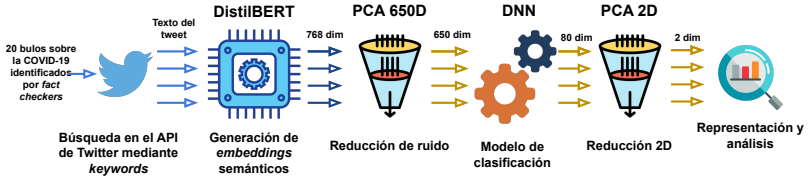


Fig. 1: Diagrama que muestra la metodología seguida, empezando con la extracción de los tuits mediante la API de Twitter, la generación de *embeddings* de las frases con DistilBERT, la reducción de dimensionalidad tras aplicar ACP, el modelo de red neuronal profunda y, finalmente, la representación reducida a dos dimensiones.

ID	Bulo	Fact checker
0	Las vacunas de ARN mensajero puedan hacer que seamos un ser transgénico	Maldita.es
1	Las vacunas contra la COVID-19 causan convulsiones	Newtral.es
2	Estados Unidos admitió que sólo un 6% de las muertes informadas fueron realmente por coronavirus	Chequeado
3	Las mascarillas causan enfermedades neurodegenerativas	Maldita.es
4	Una imagen de una patente en Países Bajos de un método para «testear el COVID-19» desde 2015	Newtral.es
5	La vacuna contra el coronavirus te puede dejar estéril	Chequeado
6	Hay un plan diseñado para el COVID-19 desde 2017 en documentos del Banco Mundial	Mala Espina Check
7	Se ha descubierto que la vacuna contra la COVID-19 destruya nuestro sistema inmunológico de forma permanente	Maldita.es
8	Beber mucha agua y hacer gárgaras con agua caliente y sal elimina el coronavirus	AFP
9	Se recomienda mantener el cuerpo en un estado alcalino.	Ecuador Chequea
10	El eucalipto previene o elimina el nuevo coronavirus	AFP
11	La hoja del árbol de guayaba puede prevenir o revertir los efectos de COVID-19	Maldita.es
12	La Nasa catalogó al dióxido de cloro como antídoto universal en 1988	Animal Político
13	Tomar vino puede ser beneficioso frente al COVID-19	EFE Verifica
14	El uso de la mascarilla provoca muertes por neumonía bacteriana	Maldita.es
15	La vitamina C previene el virus	Maldita.es
16	Christine Lagarde dijo: Los ancianos viven demasiado y eso es un riesgo para la economía global.	Chequeado
17	Existe una relación entre el laboratorio biológico chino de Wuhan las compañías farmacéuticas Glaxo y Pfizer y personas como George Soros y Bill Gates entre otros	Maldita.es
18	El coronavirus muere a los 27º.	Convoca
19	El científico Charles Libier fue detenido por crear el coronavirus Covid-19.	Animal Político

TABLE I: Lista de bulos en español obtenida de diferentes *fact-checkers.*

Para esta investigación, nos interesa el título de cada pieza de desinformación en el set de datos.

Para algunos títulos, hemos creado una *query* para la búsqueda automática en Twitter que trata de limitar la búsqueda a los *posts* que solo se refieren a esa desinformación. En este sentido, hemos optimizado la *query* con sinónimos y expresiones similares para encontrar el máximo número de tuits sobre tal falsedad en los diferentes países hispanos.

El producto de esta recolección es una base de datos con 5.071 tuits pertenecientes a 3.912 cuentas, asociados todos ellos a través de un ID a la *query* que los localizó en la red social con la búsqueda automática. Solo se han incluido los tuits en nuestra propia base de datos si: 1. La *query* recoge muchos resultados a lo largo del tiempo; 2. Al menos algunos de estos resultados sí tienen que ver con el título del bulo de LatamChequea Coronavirus; 3. No todos los tuits obtenidos pertenecen solo a las verificaciones de ese bulo por los *fact-checkers*, que serían ya la respuesta para combatir esa información falsa y no la información falsa que queremos almacenar.

C. Generación de embeddings semánticos

Antes de extraer los *embeddings* semánticos de cada tuit extraído, este se preprocesa para eliminar cualquier ruido que empeore su representación. Por tanto, hemos eliminado del texto las menciones con @ a otros usuarios, el símbolo # del *hashtag*, enlaces, guiones bajos y saltos de línea. Tras preprocesar todos los tuits, usamos el *transformer* pre-entrenado multilingüe de DistilBERT [17] para generar el *embedding* semántico de cada tuit. A partir de aquí, extraemos

el resultante vector de *embedding* para cada tuit observando el token *CLS* de la última capa de la arquitectura. Este token representa la frase entera, es decir, obteniendo, por lo tanto, 768 *features* que representan una variedad de características y que permiten calcular cómo de similares semánticamente son dos textos o entrenar un modelo de clasificación.

Como estos *embeddings* lo forman características muy diferentes que no permiten extraer una representación significativa en nuestro experimento, aplicamos además un modelo de red neuronal profunda (*DNN*, siglas del anglicismo *Deep Neural Network*) para obtener una nueva representación de los *embeddings* que aumenta la separación entre etiquetas (tuits con el mismo ID asignado para esta investigación). Sin embargo, para esto necesitamos primero eliminar las *features* redundantes que añaden ruido al modelo de clasificación, limitando su capacidad para generalizar en el set de test. Para este objetivo, aplicamos un Análisis de Componentes Principales (ACP) para reducir el número de dimensiones de los *embeddings*. Esta reducción se ha implementado experimentalmente a 650 *features*, que permiten evitar el sobreajuste sin afectar su funcionamiento. El modelo de ACP se ha ajustado en un set de entrenamiento compuesto por el 50% de los tuits.

D. Modelo de clasificación con red neuronal profunda

Para crear una representación más útil, hemos entrenado una red neuronal profunda con una única capa con 80 neuronas y activación *relu*, una capa de normalización por lotes, una capa de dilución con el 50% de probabilidad y finalmente una capa de salida con activación *softmax*. Esta arquitectura es entrenada con el 50% de los *embeddings* con el optimizador *Adam*, con el fin de aumentar la separación entre bulos no relacionados entre sí a la vez que juntamos más aquellos tuits con información falsa similar.

E. Representación de los datos

Una vez que la red neuronal ha sido entrenada, la representación de cada vector de entrada se ha extraído a partir del resultado de la capa oculta de la red neuronal profunda. Esto da lugar a vectores de 80 dimensiones, con separación aumentada entre etiquetas. Bajo la idea de representar cada tuit en un espacio bidimensional, hemos entrenado un modelo de ACP para reducir todos los *embeddings* de los tuits a dos dimensiones.

III. Resultados y discusión

El resultado de aplicar nuestra metodología de investigación explicada en las secciones previas se muestra en Fig. 2. Cada punto en el gráfico representa un tuit, mientras que el color de los puntos se refiere a cada bulo específico. Además, la posición del *embedding* del bulo en sí documentado por el *fact-checker*, el cual da origen a la creación de las *queries* y que se puede consultar en la tabla I, también se plasma. Como puede apreciarse, hay pocos bulos claramente separados cuando otros presentan grandes similitudes por su representación tan cercana en el gráfico. Mientras que los agrupamientos correspondientes a los IDs 2, 5 y 16 pueden distinguirse fácilmente de los otros, el resto de grupos están, en general, superpuestos. Esto significa que los tuits que se refieren al bulo categorizado por su ID son más diferentes en cuanto a la semántica que el resto, aunque no podamos mostrar cuáles son las propiedades que producen tales diferencias.

Por un lado, el grupo con tuits con el bulo número 2 (ID 2) se refiere a información explícita sobre el coronavirus, pero quizás difiera del resto en el tipo de contenido: mientras que la COVID-19 se menciona en el grupo 2 para difundir la falsedad de que solo un porcentaje pequeño muere en EE.UU. por esta pandemia, otros grupos ofrecen falsos remedios o conspiraciones sobre el origen del virus. Esto sugiere que en el futuro se podrían añadir más bulos de rechazo a la pandemia y que estén más cercanos al bulo del grupo 2.

Por otro lado, el grupo asociado al ID 16 podría mostrar una anomalía como la del grupo 2 por su distinta naturaleza. En este caso, los tuits que mencionan que la actual presidenta del Banco Central Europeo, Christine Lagarde, considera que los ancianos “viven demasiado” no se refieren al coronavirus pero responsabilizan a Lagarde de las víctimas de la COVID-19 a partir de una cierta edad. No obstante, los *embeddings* quizás capturen esta conexión con los virus al completo, excepto en aquellos tuits que mencionan la relación, falsa, entre sus palabras y la pandemia.

Sin embargo, el bulo 5 en sí, tal y como aparece registrado por el *fact-checker* en la tabla I, rechaza el peligro de la esterilidad causada por la vacuna, en la línea de los bulos 0, 1 y 7 plasmados en tal listado, sobre los riesgos falsos de la vacuna. A pesar de que la similitud del bulo 0 documentado por el *fact-checker* parece clara por su proximidad, esto no es tan evidente con los números 1 y 7. Esto podría indicar que los subtemas (por ejemplo, la esterilidad) dentro del gran tema (por ejemplo, la vacunación como tema principal del subtema de la esterilidad) también jugarían un gran papel en la representación final del significado en el espacio latente.

Por último, si miramos los bulos del listado correspondientes a los grupos 4, 12 y 17, muy cercanos entre sí, podríamos ver cómo los uniría la conspiración común contra instituciones científicas (una patente en Países Bajos, una afirmación de la NASA sobre el dióxido de cloro y la relación entre Wuhan con Glaxo y Pfizer, respectivamente). Además, los bulos con estos números se refieren a actividades planeadas en el pasado (la patente de 2015, la afirmación en 1988 y la relación antes de la expansión de la pandemia, respectivamente).

No obstante, la proximidad del bulo documentado por el *fact-checker* que motiva el grupo 9 sobre el mantenimiento de los cuerpos en un estado alcalino para la autodefensa anima a pensar que los temas podrían estar únicamente cerca por el peso semántico de componentes y entidades en relación con la ciencia. De hecho, las falsedades 6 y 19 de la lista de bulos también cumplen las condiciones del párrafo anterior, pero están más distantes que el resto.

Es importante resaltar la variable temporal que hemos incluido en nuestros análisis. En Fig. 3, la distribución temporal de los tuits coloreados por el ID muestra la necesidad de aplanar las diversas curvas de la infodemia. De nuevo, destaca

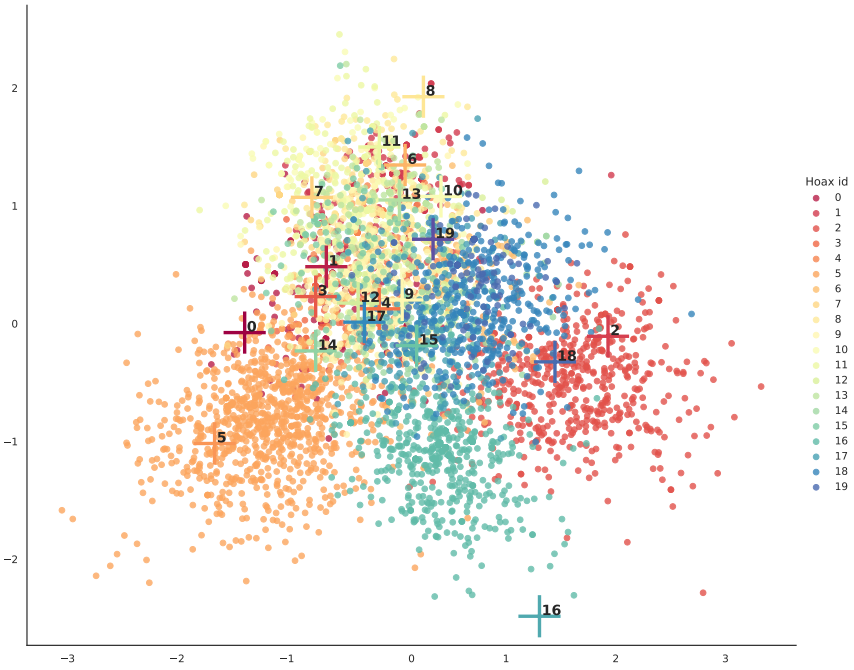


Fig. 2: Representación de los *embeddings* extraídos de la capa oculta tras aplicar Análisis de Componentes Principales para reducir los *embeddings* a un espacio bidimensional. Cada punto muestra un tuit; cada color, un bulo. Una marca y un número determinan la posición del *embedding* de cada bulo tal como está presentado por el *fact-checker* en la tabla I.

el bulo número 5 como ejemplo particular acerca del riesgo de esterilidad por la vacuna. En comparación a otros bulos con un impacto grande pero corto en el tiempo, como el número 19 (sobre el científico Charles Lieber falsamente arrestado por haber creado el virus), el grupo con el ID 5 tuvo una segunda ola con más tuits que la primera, acorde con nuestra búsqueda automática en Twitter, y ninguna de ellas fue completamente aplanada en 2020.

Una explicación de por qué este bulo fue resistente hasta el final de 2020 puede ser el tema que trata. Los bulos 0, 1, 5 y 7, referidos a las vacunas, crecieron de acuerdo al gráfico en el último trimestre de 2020 o en 2021, probablemente como respuesta a la llegada de la vacuna. La información falsa con el ID 0 también permanece a lo largo del tiempo pero, en este caso, la curva está más aplanada que el grupo número 5. Dadas estas dinámicas entre bulos, serían relevantes más experimentos que evalúen los *embeddings* semánticos con la variable temporal, para entender así el comportamiento de esta infodemia.

IV. Conclusiones

Nuestra investigación ha intentado representar la relación entre tuits que difunden bulos, desmentidos previamente en LatamChequea Coronavirus, como *embeddings* semánticos de forma visual a través de ACP y una red neuronal profunda para favorecer la separación de tuits que se refieren a una desinformación distinta. Los tuits se han agrupado en un ID que indica el bulo que contienen y que da color a los puntos en el espacio latente en dos dimensiones. También se han representado visualmente los bulos que han servido como base para la búsqueda de las publicaciones sobre esas falsedades en Twitter.

El *scatterplot* final ha demuestra cómo una agrupación de bulos junto al bulo documentado por el *fact-checker* como punto de partida está lejos o cerca de otros grupos dependiendo de su similitud entre *embeddings*. Sin embargo, aunque nuestra explicación intenta encontrar las posibles razones por las que ciertos grupos están juntos o separados, especificar las propiedades que causan la distancia entre dos bulos constituye una tarea más compleja, como hemos visto en ejemplos donde la relación o disparidad con otros grupos de información falsa no está clara.

Con todo, hemos alcanzado nuestro objetivo pero estamos aún lejos de mostrar una imagen clarificadora de la desinformación (planeada o no) y su comportamiento dentro de la infodemia. En la línea de Ng y Carley [15], la investigación futura en este campo podría orientarse a agrupar los bulos en categorías más genéricas como “Conspiración”, “Curas”, “Vacunas” o “Respuestas de Salud Pública” para entender las relaciones entre falsedades mejor. Precisamente LatamChequea Coronavirus usa este tipo de etiquetas para los bulos reportados, pero este enfoque se ha descartado para estos experimentos porque las etiquetas no son uniformes entre las noticias verificadas en tal *dataset* sobre el mismo asunto.

Agradecimientos

Este trabajo ha sido financiado por los siguientes proyectos y agencias: Ministerio de Ciencia e Innovación TIN2017-

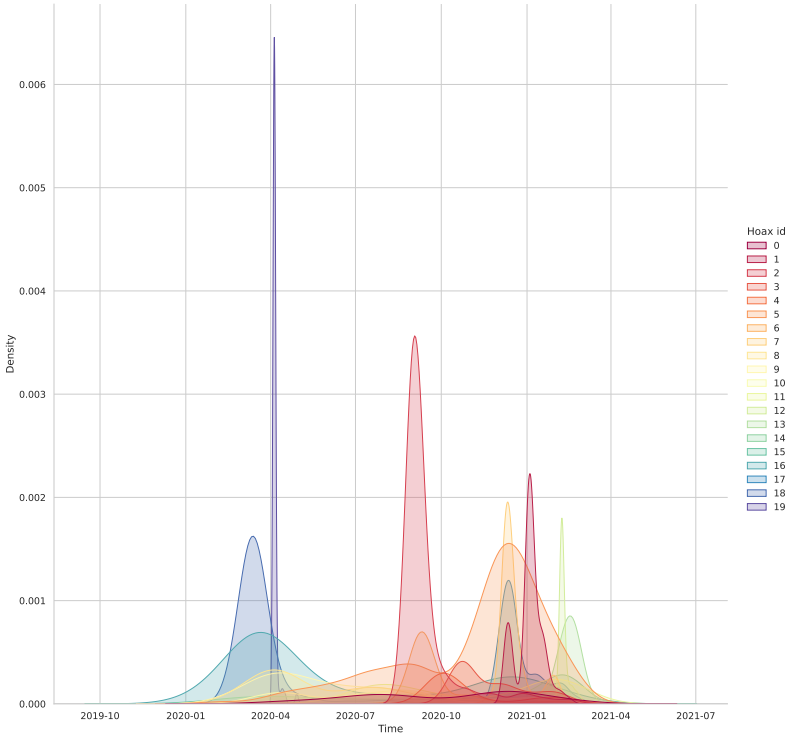


Fig. 3: Representación en el tiempo de los tuits sobre desinformación de la COVID-19. El eje x muestra las fechas de publicación; el eje y, la cantidad de publicaciones. Cada color representa un bulo.

85727-C4-3-P (DeepBio), Comunidad de Madrid S2018/TCS-4566 grant (CYNAMON), Fundación BBVA, Ayudas a Equipos de Investigación Científica sobre SARS-CoV-2 y COVID-19 en Economía y Ciencias Sociales en el marco del proyecto "CIVIC: Caracterización inteligente de la veracidad de información asociada a la COVID-19", por la Comisión Europea, proyecto 2020-EU-IA-0252 IBERIFIER - Iberian Digital Media Research and Fact-Checking Hub. Acción parcialmente financiada por la Comunidad de Madrid en el marco del Convenio Plurianual con la Universidad Politécnica de Madrid en la línea de actuación Programa de Excelencia para el Profesorado Universitario.

REFERENCES

[1] J. S. Brennen, F. Simon, P. N. Howard, and R. K. Nielsen, "Types, sources, and claims of covid-19 misinformation," *Reuters Institute*, vol. 7, pp. 3–1, 2020.

[2] T. T. Nguyen, "Artificial intelligence in the battle against coronavirus (covid-19): a survey and future research directions," *arXiv preprint arXiv:2008.07343*, 2020.

[3] C. Shorten, T. M. Khoshgoftaar, and B. Furht, "Deep learning applications for covid-19," *Journal of Big Data*, vol. 8, no. 1, pp. 1–54, 2021.

[4] M. Choraś, K. Demestichas, A. Giełczyk, Á. Herrero, P. Ksieniewicz, K. Remoundou, D. Urda, and M. Woźniak, "Advanced machine learning techniques for fake news (online disinformation) detection: A systematic mapping study," *Applied Soft Computing*, p. 107050, 2020.

[5] O. Gencoglu, "Large-scale, language-agnostic discourse classification of tweets during covid-19," *Machine Learning and Knowledge Extraction*, vol. 2, no. 4, pp. 603–616, 2020.

[6] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.

[7] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," *arXiv preprint arXiv:1310.4546*, 2013.

[8] D. Jurafsky and J. H. Martin, "Speech and language processing (3rd draft ed.)," 2019.

[9] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," *arXiv preprint arXiv:1802.05365*, 2018.

[10] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[11] M. Asgari-Chenaghlu, N. Nikzad-Khasmakhi, and S. Minaee, "Covid-transformer: Detecting covid-19 trending topics on twitter using universal sentence encoder," 2020.

[12] N. Chen, Z. Zhong, and J. Pang, "An exploratory study of covid-19 information on twitter in the greater region," *Big Data and Cognitive Computing*, vol. 5, no. 1, p. 5, 2021.

[13] T. Raha, V. Indurthi, A. Upadhyaya, J. Kataria, P. Bommakanti, V. Keswani, and V. Varma, "Identifying covid-19 fake news in social media," *arXiv preprint arXiv:2101.11954*, 2021.

[14] A. Wani, I. Joshi, S. Khandve, V. Wagh, and R. Joshi, "Evaluating deep learning approaches for covid19 fake news detection," *arXiv preprint arXiv:2101.04012*, 2021.

[15] L. H. X. Ng and K. M. Carley, "The coronavirus is a bioweapon: Analysing coronavirus fact-checked stories," *arXiv preprint arXiv:2104.01215*, 2021.

[16] LatamChequea, "Información chequeada sobre el coronavirus," 2020. [Online]. Available: https://chequeado.com/latamcoronavirus/

[17] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter," *arXiv*, Oct 2019. [Online]. Available: https://arxiv.org/abs/1910.01108v4

Mejorando Redes Profundas para Aprendizaje por Transferencia mediante Capas Dispersas optimizadas con Algoritmos Evolutivos

Javier Poyatos
Dpto Ciencias de Datos e IA, Instituto Interuniversitario de Investigación en Data Science and Computational Intelligence (DaSCI), Universidad de Granada, España
jpoyatosamador@ugr.es

Daniel Molina
Dpto Ciencias de Datos e IA, Instituto Interuniversitario de Investigación en Data Science and Computational Intelligence (DaSCI), Universidad de Granada, España
dmolina@decsai.ugr.es

Aritz D. Martinez
TECNALIA, Basque Research & Technology Alliance (BRTA)
Derio, España
aritz.martinez@tecnalia.com

Siham Tabik
Dpto Ciencias de Datos e IA, Instituto Interuniversitario de Investigación en Data Science and Computational Intelligence (DaSCI), Universidad de Granada, España
siham@ugr.es

Javier Del Ser
TECNALIA, Basque Research & Technology Alliance (BRTA)
Universidad del País Vasco (UPV/EHU)
Derio, España
javier.delser@tecnalia.com

Francisco Herrera
Dpto Ciencias de Datos e IA, Instituto Interuniversitario de Investigación en Data Science and Computational Intelligence (DaSCI), Universidad de Granada, España
herrera@decsai.ugr.es

***Resumen*—En los últimos años, las redes neuronales profundas han mostrado un buen rendimiento en problemas de modelado complejos. Un problema aun por resolver es el diseño de la arquitectura óptima para dichos modelos. Afortunadamente, el aprendizaje por transferencia ayuda a resolver este problema incorporando conocimiento previo de otros problemas. En este contexto, la configuración de las capas totalmente conectadas es crucial. Este trabajo propone EvoPodaAT, un algoritmo evolutivo que reemplaza dichas capas por capas dispersas que son optimizadas usando un algoritmo genético. Ejecutamos EvoPodaAT sobre diferentes conjuntos de datos disponibles en la literatura para analizar la estrategia de codificación para representar las capas evolucionadas. Los resultados indican que EvoPodaAT es capaz de mejorar la precisión de las predicciones en todos los conjuntos de datos, al mismo tiempo que reduce significativamente el número de neuronas activas en las últimas capas, contribuyendo así a la eficiencia computacional de la red.**

***Index Terms*—Aprendizaje Profundo, Algoritmos Evolutivos, Algoritmo Genéticos, Aprendizaje por Transferencia, Poda.**

I. Introducción

El *Aprendizaje Profundo* [1], AP, se ha convertido en los últimos años en una de las áreas de investigación más atractivas del aprendizaje automático. Sus modelos han ofrecido buenos resultados en una amplia variedad de aplicaciones tales como procesamiento de lenguaje natural y la detección de objetos.

Un tipo de problemas para los que más se ha aplicado el AP por su excelente resultado es la clasificación de imágenes. Los modelos de AP se comportan como extractores de características y, después, un conjunto de capas que asocian estas características a las clases. Dichas capas suelen ser totalmente conectadas, también llamadas densas, en el que cada neurona de cada capa está conectada con todas las neuronas de la capa anterior. Este proceso suele implicar el entrenamiento de millones de parámetros, con lo que eso supone en tiempo de aprendizaje. Una alternativa es el uso del *Aprendizaje por Transferencia*, AT [2], que permite importar el conocimiento de otro problema que esté relacionado con el actual, lo cual reduce el tiempo de entrenamiento. En particular, es común el uso de modelos de AP pre-entrenados con un conjunto de datos (como ImageNet [3]) y añadir, tras ello, capas densas para adaptar la red al problema [4]. En este enfoque de AT únicamente se necesita aprender/entrenar las capas densas finales para obtener excelentes resultados. Sin embargo, la arquitectura de estas capas es una decisión crítica y su diseño en cuanto al número de capas y de neuronas por capa es un problema abierto en la literatura [5].

Los Algoritmos Evolutivos [6], AEs, son algoritmos populares que se pueden aplicar a todo tipo problemas complejos de optimización, ya que proporcionan buenos resultados en tiempo razonable y con recursos limitados. Uno de los más

populares son los algoritmos genéticos [7], AGs, cuya versatilidad permite la optimización en diversas áreas [8].

Los AEs son capaces de optimizar redes neuronales [9], pero recientemente se están usando en redes de tipo AP [10]. Se han utilizado los AEs para reducir de la complejidad de la red neuronal mediante la desconexiones neuronas y conexiones, dentro de un proceso que se conoce como *poda* [11], creando nuevas redes con menor coste computacional para el entrenamiento. A diferencia de trabajos previos, como [12], cuyo principal objetivo es reducir la complejidad de la red, nuestro trabajo se enfoca en adaptar únicamente las capas densas, principalmente para mejorar la precisión del modelo usando AT (la reducción es menor ya que solamente nos centramos en las capas densas).

En este artículo proponemos *Algoritmo evolutivo de poda para el aprendizaje por transferencia*, EvoPodaAT, que reemplaza las capas densas de los modelos de AP usando AT por capas dispersas que son optimizadas mediante un AG. El nuevo diseño obtenido por EvoPodaAT permite mejorar la precisión del modelo.

Hemos analizado EvoPodaAT con distintos conjuntos de datos para demostrar la mejora en precisión que es capaz de obtener reduciendo la complejidad de los mismos. Además, comparamos EvoPodaAT con varios modelos de capas densas y métodos de poda de la literatura, observando que EvoPodaAT obtiene mejores resultados.

El resto del artículo está estructurado de la siguiente manera: la Sección II describe en detalle EvoPodaAT. La Sección III presenta los detalles del marco experimental. En la Sección IV se muestran y debaten los resultados de los experimentos realizados. Finalmente, en la Sección V exponemos las principales conclusiones y trabajo futuro.

II. Propuesta: Algoritmo evolutivo de poda para el aprendizaje por transferencia

Esta sección describe la propuesta EvoPodaAT. Primero introducimos el concepto de capa dispersa. Posteriormente, definimos el algoritmo genético que lo optimiza. Por último, se describe el proceso de creación de la red y realiza el proceso de poda de la misma.

II-A. Esquema global

La propuesta busca mejorar el rendimiento de una red neuronal reduciendo el número de neuronas activas. Para ello, usamos una capa dispersa, que está compuesta por un subconjunto del total de conexiones de una capa densa.

EvoPodaAT usa un AG estacionario (*steady state*) específicamente diseñado para optimizar las conexiones de la capa dispersa. La capa dispersa se configura mediante una máscara de valores booleanos que indican, para cada conexión, si se mantiene activa o desactivada (una conexión desactivada mantiene su peso igual a cero). El AG usado presenta una representación binaria: Cada individuo funciona como una máscara para la red neuronal y, a partir de él, se crea una capa dispersa en lugar de la capa densa.

La optimización de la capa dispersa se puede hacer a nivel de conexiones o a nivel de neuronas. En nuestro modelo, cada gen del cromosoma evolucionado por el AG representa el número de neuronas activas. El valor 1 en la posición i indica que dicha neurona está activa, mientras que 0 significa que está desactivada, es decir, todas sus conexiones están eliminadas tanto en entrenamiento como en evaluación. La longitud de cada cromosoma es, por tanto, el número de neuronas de la capa dispersa. Un ejemplo de la estrategia de codificación se muestra en la Fig. 1.

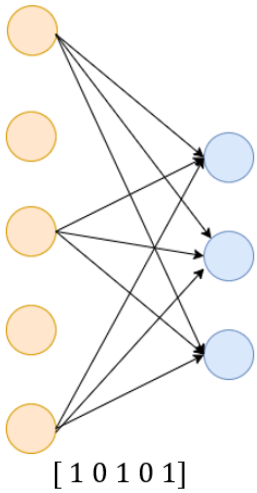


Figura 1: Representación de una solución.

II-B. Elementos de EvoPodaAT

El componente evolutivo es un AG estacionario en el que en cada iteración se cruzan dos individuos para generar dos nuevos, que se introducen en la población si su calidad es mejor que la de los peores de la población vigente. Como hemos comentado anteriormente, en EvoPodaAT cada cromosoma es un vector binario y cada gen representa una conexión entre dos capas. Cada generación comprende el clásico esquema de selección, cruce, mutación y reemplazo. Las mejores soluciones encontradas en el proceso evolutivo se guardan en la población de individuos. A continuación, describimos los diferentes componentes:

Selección: el operador de selección implementado es *Negative Assorting Mating, NAM* [13]. El primer padre se toma de forma uniformemente aleatoria, mientras que el segundo es escogido entre tres posibles candidatos. El candidato con la mayor distancia de Hamming respecto al primer padre es el segundo padre. Este método de selección permite un mayor grado de exploración del espacio de búsqueda.

Cruce: EvoPodaAT usa el operador de cruce uniforme. Dados dos padres **P** y **Q** se genera un nuevo individuo **S**, para cada gen se genera aleatoriamente un número $r_i \in [0.0, 1.0]$, y para la posición $S_i = B_i$ si $r_i \leq 0.5$, y $S_i = Q_i$ si $r_i > 0.5$.

Mutación: EvoPodaAT realiza la mutación con una probabilidad p_m para cada individuo: cada vez que un individuo se

muta, se elige un gen de forma uniformemente aleatoria y se invierte su valor.

Estrategia de reemplazo: tras cada generación los individuos resultantes tras los operadores genéticos reemplazan a los dos peores elementos de la población si los mejoran en cuanto a calidad (fitness).

Inicialización: para cada individuo se asigna una probabilidad diferente p_{one} (muestreando una distribución uniforme entre 0 y 1), y sus genes se inicializan a 0 o 1 en función de dicha probabilidad:

$$I_i = \begin{cases} 1 & \text{si } r \leq p_{one}, \\ 0 & \text{en otro caso,} \end{cases} \quad (1)$$

donde r es la realización de una variable aleatoria continua uniformemente distribuida en el intervalo $[0.0, 1.0]$.

Evaluación de los individuos: para evaluar un individuo, se entrena la red neuronal codificada por dicho individuo sobre el conjunto de entrenamiento, y se toma como valor *fitness* la precisión obtenida por dicha red sobre el conjunto de test.

II-C. Configuración de red de EvoPodaAT

La propuesta se ha diseñado asumiendo que el uso de AT permite importar de otros modelos para la extracción de características, siendo ResNet 50 el modelo de CNN. Se ha adaptado dicho modelo con una red pre-entrenada sobre ImageNet. Tras ello, se han añadido una capa densa de 512 neuronas, la capa dispersa también con 512 neuronas, que es optimizada mediante EvoPodaAT. Por último, la capa de salida. con tantas neuronas como clases a discriminar.

La capa dispersa se crea en base a una matriz de adyacencia que define las conexiones entre las capas densas. La definición de esta matriz permite realizar la poda en relación a las neuronas de la primera o segunda capa densa. Estas dos maneras de realizar la poda las lleva a cabo EvoPodaAT y se describen a continuación:

- El primer método es llamado ***primera capa*** y significa que la poda se realiza en función de las N entradas de la primera capa densa (*FC-1*), eliminado sus conexiones hacia la segunda capa (*FC-2*). Este esquema se expone en la Fig. 2.

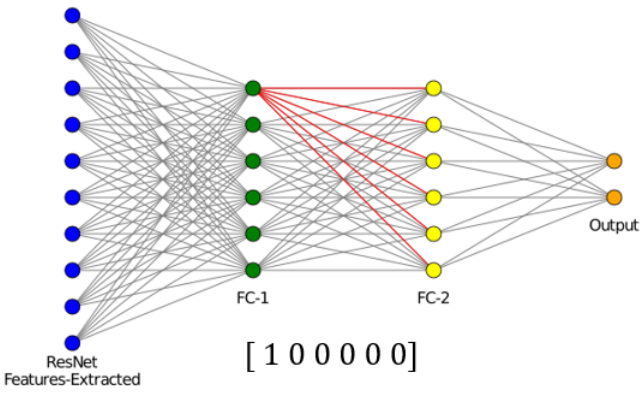


Figura 2: Representación tipo *primera capa*.

- El segundo método es denominado ***segunda capa***. Representa la poda de las conexiones entre la primera y segunda capa, pero tomando como referencia las neuronas de la segunda capa densa. Consecuentemente, puede haber neuronas de esta capa que no tengan ninguna conexión con las de la capa previa. La Fig. 3 muestra un ejemplo de este mecanismo en el que las conexiones que son eliminadas se indican en rojo si su neurona no está activa.

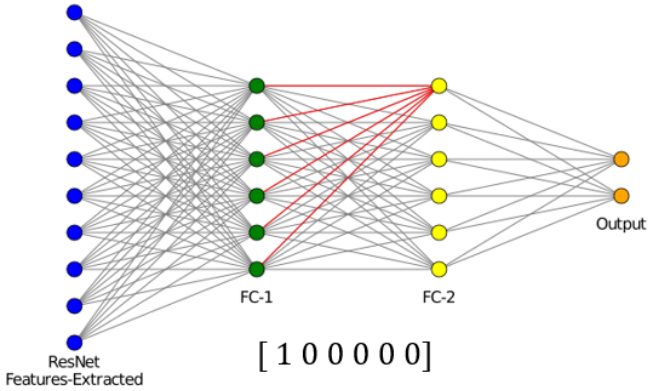


Figura 3: Representación tipo *segunda capa*.

III. Marco Experimental

En esta sección describimos los conjuntos de datos usados para examinar el desempeño práctico de nuestra propuesta, y posteriormente la configuración de entrenamiento y los parámetros de EvoPodaAT.

III-A. Conjuntos de datos

Hemos escogido diferentes conjuntos de datos adecuados para el enfoque de AT valorando que el tiempo de entrenamiento e inferencia sea bajo para permitirnos evaluar un número suficiente de individuos:

- En SRSMAS [14] se clasifican diferentes clases de arrecifes de coral, permite distinguir entre distintas clases con cierta dificultad.
- En RPS [15] se identifica el gesto de las manos en el juego infantil de *piedra, papel, o tijera,* a partir de imágenes dispuestas en diferentes posiciones y con diferentes tonalidades de piel.
- En PLANTAS [16], la tarea consiste en diferenciar entre tipos de hojas y plantas, como el tomate, manzana, maíz, entre otras.

La Tabla I describe sus características principales.

Tabla I: Conjuntos de datos usados en los experimentos.

Conjuntos de datos	Tamaño imagen	Número de clases	# Instancias (entrenamiento / test)
SRSMAS	(299, 299)	14	333 / 76
RPS	(300, 300)	3	2520 / 372
PLANTAS	(100, 100)	27	2340 / 236

III-B. Configuración de entrenamiento

Para los experimentos se divide cada conjunto de datos anterior en subconjuntos de entrenamiento y prueba. El conjunto de datos SRSMAS está definido por una validación cruzada de cincos subconjuntos, mientras que los otros dos tienen ya predefinidos sus conjuntos de entrenamiento y test.

El entrenamiento de la red se realiza usando SDG, con tamaño *batch* de 32 imágenes y un máximo de 600 épocas (aunque el entrenamiento se detiene si no hay mejora en diez épocas consecutivas). Dado que este enfoque usa AT, únicamente se entrenan las últimas capas son entrenadas, el resto de capas de la ResNet-50 mantienen los valores pre-entrenados.

Tabla II: Parámetros de EvoPodaAT.

Parámetro	**Valor**
Evaluaciones Máx.	200 (una capa) 300 (ambas capas)
# ejecuciones	5
Tam. población	30
NAM	3
p_{mut}	0.07

Los parámetros de EvoPodaAT se indican en la Tabla II. Dado el carácter estocástico de la propuesta, se ha ejecutado 5 veces por cada dataset, mostrando la media de estas ejecuciones. El algoritmo devuelve la configuración encontrada con mejor precisión (a igualdad de precisión, devuelve la que presente menor porcentaje de neuronas activas). Para cada conjunto de datos el algoritmo se ejecuta 5 veces, y se devuelve el valor promedio. Todos los experimentos se realizan usando TensorFlow 2, Python 3.6 y desplegado en una GPU Tesla V100-SXM2 GPU.

IV. Resultados y análisis

En esta sección analizamos el comportamiento de EvoPodaAT con diferentes experimentos y conjuntos de datos y para determinar la bondad de la propuesta. Primero vamos a comparar EvoPodaAT frente a modelos de referencia de capas densas, para mostrar cómo puede mejorar frente a un modelo de capa densa. Después, compararemos EvoPodaAT con métodos de poda eficientes para la compresión de redes CNN publicados en la literatura. Evaluaremos tanto el acierto de la red como la compresión del modelo. En las distintas tablas los mejores resultados se resaltan en negrita.

IV-A. Comparación de EvoPodaAT con modelos de referencia

En esta sección queremos determinar la diferencia de rendimiento entre EvoPodaAT y modelos de referencia descritos a continuación:

- El primer modelo presenta dos capas densas de 512 neuronas y la capa de salida. Este modelo equivale a un modelo cuyos genes tienen valor 1 en todos los casos. Este modelo es denominado *Sin Poda*.
- Una búsqueda en malla, que permite determinar si la mejora obtenida por EvoPodaAT se puede obtener con una simple búsqueda sobre el porcentaje de neuronas de la capa densa. Se han probado diferentes modelos con porcentajes desde un 10 % hasta 100 % con incremento de 10 %. El mejor modelo encontrado en función de su precisión se denota como *Mejor Densa FC-X*, donde X puede tomar el valor 1 o 2, en función si se aplica la poda a la *primera capa* o a la *segunda capa*. Denotaremos el caso de la optimización por dos capas como *Mejor Densa Ambas*.

Primero comparamos el método de *primera capa* con los modelos descritos en la Sección IV-A. La Tabla III muestra los resultados de los experimentos.

Tabla III: EvoPodaAT respecto a los modelos sin poda y modelos fijos en el enfoque de tipo *primera capa*.

Conjunto de datos	Medida	Sin doda	Mejor densa FC-1	EvoPodaAT
SRSMAS	Precisión	0.858	0.858	**0.883**
	% Activas neur.	100	100	51
RPS	Precisión	0.922	0.938	**0.964**
	% Activas neur.	100	30	22
PLANTAS	Precisión	0.402	0.448	**0.482**
	% Activas neur.	100	10	53

EvoPodaAT obtiene un mejor rendimiento para todos los casos en cuanto a precisión, pero el caso de PLANTAS es el único donde la referencia tiene un mejor porcentaje de reducción. Como conclusión, EvoPodaAT es capaz de mejorar el rendimiento de la red y, al mismo tiempo, reducir el número de neuronas activas, aprendiendo las conexiones importantes para la obtención de esta mejora.

Ahora comparamos *segunda capa* con los modelos de referencia, siguiendo un razonamiento análogo al caso anterior. La Tabla IV muestra los resultados de este segundo experimento.

Tabla IV: EvoPodaAT respecto a los modelos sin poda y modelos fijos en el enfoque de tipo *segunda capa*.

Conjunto de datos	Medida	Sin poda	Mejor densa FC-2	EvoPodaAT
SRSMAS	Precisión	0.858	0.860	**0.884**
	% Activas neur.	100	80	47
RPS	Precisión	0.922	0.949	**0.969**
	% Activas neur.	100	30	16
PLANTAS	Precisión	0.402	0.441	**0.483**
	% Activas neur.	100	50	37

EvoPodaAT supera en precisión y porcentaje de neuronas activas a los modelos de referencia en todos los casos. Además, los resultados son ligeramente mejores que los correspondientes al caso de la primera capa. Como conclusión, verificamos que ambos enfoques mejoran a los respectivos modelos de referencia y que el enfoque segunda capa obtiene el mejor rendimiento.

Por último, realizamos la poda en dos capas consecutivas usando el enfoque *segunda capa*. La configuración de capas densas es la misma que en los casos anteriores, pero nos centramos en las dos capas a la vez. El número de evaluaciones para estos experimentos es 300, incrementando en 100

evaluaciones respecto al empleado en las experimentaciones previas, debido a que el espacio de búsqueda es ahora más extenso.

La Tabla V muestra los resultados de EvoPodaAT para estos experimentos. El modelo *Mejor Densa* explora todas las posibles combinaciones de 10 % a 100 % de neuronas entre las dos capas densas. Lo resultados reportan una mejora en precisión y reducción del modelo en todos los conjuntos de datos, salvo en SRSMAS, donde la reducción no supera al mejor modelo de referencia.

Tabla V: EvoPodaAT respecto a los modelos sin poda y mejor densa en el enfoque de ambas capas.

Conjunto de datos	Medida	Sin poda	Mejor densa ambas	EvoPodaAT
SRSMAS	Precisión	0.858	0.863	**0.885**
	% Activas neur.	100	50	64
RPS	Precisión	0.922	0.946	**0.978**
	% Activas neur.	100	90	12
PLANTAS	Precisión	0.402	0.466	**0.491**
	% Activas neur.	100	55	41

La Tabla VI muestra el resumen de los experimentos tanto el mejor como en media de las diferentes propuestas. Los resultados nos muestran que la poda de las dos capas obtiene en media los mejores resultados de precisión en los conjuntos de datos. En cuanto a la reducción del modelo, no hay una tendencia clara que se observe en los resultados, si bien optimizar la segunda capa es lo que presenta una mayor reducción en la mayoría de los casos considerados.

Tabla VI: Resultados de EvoPodaAT para cada nivel de poda.

Conjunto de datos	Nivel Poda	Mejor EvoPodaAT		Media EvoPodaAT	
		Precisión	% Activas	Precisión	% Activas
SRSMAS	FC-1	**0.889**	35.74	0.883	50.70
	FC-2	**0.889**	36.13	0.884	46.64
	Both (200 iter.)	0.884	52.54	0.884	64.18
	Both (300 iter.)	**0.889**	75.98	**0.885**	64.26
RPS	FC-1	**0.989**	1.56	0.964	22.11
	FC-2	**0.989**	1.76	0.969	15.90
	Both (200 iter.)	0.981	3.12	0.972	12.13
	Both (300 iter.)	0.984	3.42	**0.978**	12.23
PLANTAS	FC-1	0.491	58.98	0.482	52.81
	FC-2	**0.496**	33.59	0.482	37.26
	Both (200 iter.)	0.491	30.27	0.485	41.48
	Both (300 iter.)	0.491	30.27	**0.491**	41.35

Estos resultados nos permiten hacer una reflexión sobre los beneficios de EvoPodaAT: mejora el rendimiento de los modelos de referencia y, además, es capaz de reducir el número de neuronas de ambas capas en el proceso de optimización.

En conclusión, nuestros resultados demuestran que EvoPodaAT es capaz de mejorar el desempeño de los modelos de referencia y, al mismo tiempo, reducir el número de neuronas tanto en los enfoques de una capa como en el de dos capas.

IV-B. Comparando con otros métodos de poda

Esta sección tiene como objetivo la comparación de EvoPodaAT con otros métodos de poda de CNN, tanto en función de la precisión como de la complejidad computacional. Estos métodos necesitan conocer el porcentaje objetivo de poda, siendo éste el que se obtiene en el proceso de optimización por EvoPodaAT. Los métodos implementados son los siguientes:

- **Pesos** [17]: los parámetros con menor peso son los eliminados. Este método opera sobre el conjunto total de parámetros de la capa a optimizar.
- **Decaimiento Polinómico** [18]: los parámetros se eliminande acuerdo a un decaimiento polinómico (*polynomial decay*) especificado por el número de neuronas activas finales de la red. Se permite que la red se entre durante 5 épocas cada vez que se eliminan conexiones. Este enfoque se realiza también sobre la capa a optimizar.
- **Neuronas en Primera Capa** [19]: las neuronas en la primera capa que tengan los menos valores en media son podadas, tal y como se muestra en la Fig. 2.
- **Neuronas en Segunda Capa** [19]: las neuronas en la segunda capa que tengan los menos valores en media son podadas, véase la Fig. 3 para un ejemplo ilustrativo.

La Tabla VII muestra los resultados promedio de los diferentes enfoques, poniendo de manifiesto la mejora de nuestros modelos respecto a los métodos de poda con los que hemos comparado nuestra propuesta. Por último, mostramos los tiempos de ejecución de nuestra EvoPodaAT respecto a los modelos de poda en la Tabla VIII. Los resultados evidencian el efecto de la convergencia más lenta de nuestro enfoque evolutivo, ya que necesita más tiempo para completar su exploración.

V. Conclusiones

Este trabajo presenta EvoPodaAT, un nuevo algoritmo evolutivo de poda para adaptar la arquitectura de las últimas capas en un modelo de aprendizaje profundo. EvoPodaAT comienza con una red inicializada previamente aprendida en una tarea relacionada con la que se desea resolver, optimizando las capas densas para determinar qué neuronas deben ser desactivadas para una mejor adaptación al problema. El uso de EvoPodaAT permite obtener una configuración específica para cada conjunto de datos mediante la evolución de la capa dispersa mediante un AG especialmente diseñado para este problema, eliminando neuronas innecesarias.

Se ha llevado a cabo una experimentación amplia para comprobar la bondad de EvoPodaAT. Los resultados muestran que nuestra propuesta mejora la precisión obtenida por los modelos en cada conjunto de datos frente al uso de capas sin podar. Además, hemos constatado que la poda de dos capas consecutivas proporciona mejores resultados en media que un enfoque que considera una única capa. Adicionalmente, las redes obtenidas han reducido el número de neuronas activas. Finalmente, se ha comparado con otros métodos de poda de la literatura y los resultados muestran que ha obtenido mejor porcentaje en precisión que dichos métodos. EvoPodaAT ha obtenido un mejor rendimiento de la red que los modelos de la literatura, pese a que nuestro método es más lento en cuanto a la convergencia. Por lo tanto, podemos concluir que EvoPodaAT es capaz de determinar qué conexiones son las mejores con conocimiento importado de otros problemas relacionados.

Tabla VII: Resultados en media de EvoPodaAT frente a métodos eficientes de poda en los enfoques *primera capa* y *segunda capa*.

		Primera capa				**Segunda capa**				**Ambas capas**			
Conjunto de datos	Medida	Pesos	Decai. polin.	Neuronas primera	EvoPodaAT	Pesos	Decai. polin.	Neuronas segunda	EvoPodaAT	Pesos	Decai. polin.	Neuronas segunda	EvoPodaAT
SRSMAS	Precisión	0.835	0.842	0.788	**0.883**	0.834	0.837	0.779	**0.884**	0.845	0.847	0.647	**0.885**
	% Activas neur.	51	51	51	51	47	47	47	47	64	64	64	64
RPS	Precisión	0.886	0.914	0.853	**0.964**	0.845	0.911	0.696	**0.969**	0.694	0.899	0.490	**0.978**
	% Activas neur.	22	22	22	22	15	15	15	15	12	12	12	12
PLANTAS	Precisión	0.398	0.418	0.385	**0.482**	0.392	0.420	0.313	**0.482**	0.393	0.411	0.278	**0.491**
	% Activas neur.	53	53	53	53	37	37	37	37	41	41	41	41

Tabla VIII: Tiempo en segundos de EvoPodaAT frente a métodos de poda eficientes en los diferentes enfoques.

	Primera o Segunda capa					**Ambas capas**			
Conjunto de datos	Pesos	Decai. polin.	Enfoque neuronas	EvoPodaAT FC-1	EvoPodaAT FC-2	Pesos	Decai. polin.	Enfoque neuronas	EvoPodaAT ambas
SRSMAS	2,395	2,545	2,395	48,733	38,731	2,395	2,545	2,395	55,596
RPS	1,229	1,378	1,229	21,440	18,790	1,229	1,379	1,229	28,774
PLANTAS	270	366	270	23,042	22,235	270	370	270	35,998

En trabajos futuros estudiaremos el uso de algoritmo evolutivos específicos para trabajar a diferentes niveles de la arquitectura neuronal del modelo AP, mejorando las conexiones activas para conseguir una optimización más rápida y detallada. También analizaremos la viabilidad de optimizar la red a nivel de conexiones, para lo cual prevemos necesario importar avances algorítmicos en AEs diseñados para problemas de elevada dimensionalidad. Por último, planeamos extender la aplicación de EvoPodaAT a otros paradigmas de aprendizaje, como la segmentación de imágenes o detección de objetos.

AGRADECIMIENTOS

F. Herrera, S. Tabik, D. Molina y J. Poyatos están respaldados por el proyecto Andalusian Excellence P18-FR-4961 y el proyecto de infraestrctura con referencia EQC2018-005084-P, concedido por el Ministerio de Ciencia e Innovación y el Fondo Europeo de Desarrollo Regional (ERDF). Aritz D. Martinez y Javier Del Ser agradecen al Gobierno Vasco el apoyo recibido a través de los programas de financiación de proyectos I+D EMAITEK y ELKARTEK, así como el grupo consolidado de investigación MATHMODE (IT1294-19).

REFERENCIAS

[1] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, http://www.deeplearningbook.org.

[2] S. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[3] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[4] H.-C. Shin, H. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura, and R. Summers, "Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics and Transfer Learning," *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1285–1298, 2016.

[5] W. Liu, Z. Wang, X. Liu, N. Zeng, Y. Liu, and F. Alsaadi, "A survey of deep neural network architectures and their applications," *Neurocomputing*, vol. 234, pp. 11–26, 2017.

[6] T. Bäck, D. B. Fogel, and Z. Michalewicz, Eds., *Handbook of Evolutionary Computation*. Bristol, UK: IOP Publishing Ltd., 1997.

[7] D. Goldberg, "Genetic Algorithms in Search, Optimization, and Machine Learning," *Addison-Wesley, Reading, MA*, 1989.

[8] L. Chambers, *The Practical Handbook of Genetic Algorithms: Applications, Second Edition*. Chapman and Hall/CRC, 2000.

[9] H. Iba, *Evolutionary Approach to Machine Learning and Deep Neural Networks: Neuro-Evolution and Gene Regulatory Networks*. Springer, 2018.

[10] A. D. Martinez, J. Del Ser, E. Villar-Rodriguez, E. Osaba, J. Poyatos, S. Tabik, D. Molina, and F. Herrera, "Lights and shadows in evolutionary deep learning: Taxonomy, critical methodological analysis, cases of study, learned lessons, recommendations and challenges," *Information Fusion*, vol. 67, pp. 161–194, 2021.

[11] Z. Wang, F. Li, G. Shi, X. Xie, and F. Wang, "Network pruning using sparse learning and genetic algorithm," *Neurocomputing*, vol. 404, pp. 247 – 256, 2020.

[12] J. Frankle and M. Carbin, "The lottery ticket hypothesis: Finding sparse, trainable neural networks," in *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9*, 2019.

[13] C. Fernandes and A. Rosa, "A study on non-random mating and varying population size in genetic algorithms using a royal road function," in *Proceedings of the 2001 Congress on Evolutionary Computation*, vol. 1. IEEE, 2001, pp. 60–66.

[14] A. Gómez-Ríos, S. Tabik, J. Luengo, A. Shihavuddin, and F. Herrera, "Coral species identification with texture or structure images using a two-level classifier based on convolutional neural networks," *Knowledge-Based Systems*, p. 104891, 07 2019.

[15] L. Moroney. (2019, feb) Rock, paper, scissors dataset. [Online]. Available: http://www.laurencemoroney.com/rock-paper-scissors-dataset/

[16] D. Singh, N. Jain, P. Jain, P. Kayal, S. Kumawat, and N. Batra, "Plantdoc: A dataset for visual plant disease detection," in *Proceedings of the 7th ACM IKDD CoDS and 25th COMAD*, 2020, p. 249–253.

[17] S. Han, J. Pool, J. Tran, and W. Dally, "Learning both weights and connections for efficient neural network," in *Advances in Neural Information Processing Systems*, vol. 28. Curran Associates, Inc., 2015.

[18] M. Zhu and S. Gupta, "To prune, or not to prune: Exploring the efficacy of pruning for model compression," in *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018*, 2018.

[19] S. Srinivas and R. V. Babu, "Data-free parameter pruning for deep neural networks," in *Proceedings of the British Machine Vision Conference 2015, BMVC 2015, Swansea, UK, September 7-10*, 2015, pp. 31.1–31.12.

XIV Congreso Español de Metaheurísticas, Algoritmos Evolutivos y Bioinspirados

Sesión especial:

Aplicaciones en Medicina y Biotecnología



Predicción de la glucosa en sangre mediante un sistema basado en reglas difusas TSK de dos fases

J. Alvarado
Universidad Complutense de Madrid
Madrid, España
jorgal07@ucm.es

J. Manuel Velasco
Universidad Complutense de Madrid
Madrid, España
mvelascc@fis.ucm.es

F. Chávez
Universidad de Extremadura
Mérida, España
fchavez@unex.es

Almudena Sánchez
Universidad Complutense de Madrid
Madrid, España
almsan09@ucm.es

F. Fernández de Vega
Universidad de Extremadura
Mérida, España
fcofdez@unex.es

***Resumen*—Llevar el control de la glucosa es una tarea compleja que las personas con diabetes suelen realizar por sí mismas. Una predicción precisa y oportuna es de vital importancia para tomar decisiones y recomendar acciones correctivas al paciente cuando el valor que va a tener la glucosa en sangre se encuentra fuera de un rango objetivo. Es crucial predecir eventos como la hiperglucemia y la hipoglucemia que pueden causar daños a la salud a corto plazo y potenciales daños permanentes a largo plazo. El objetivo de este trabajo es diseñar un modelo de predicción de los valores de glucosa en sangre utilizando un sistema basado en reglas difusas Takagi-Sugeno-Kang de dos fases. La primera fase es un proceso de aprendizaje por medio del cual las funciones de pertenencia y la base de reglas se optimizan mediante un algoritmo genético. En la segunda fase de ajuste, utilizamos nuevamente un algoritmo genético para realizar la selección y optimización de las reglas. Para entrenar nuestro modelo utilizamos dos escenarios diferentes, *What-if* y *Agnóstico*; en ambos las entradas son valores medidos por un sistema de monitorización continua de glucosa, así como la ingesta previa de carbohidratos y las inyecciones de insulina. En el escenario *What-if* se aceptan valores futuros supuestos de carbohidratos e inyecciones de insulina. Por otro lado, en el escenario *Agnóstico*, sólo se dispone de información de los eventos pasados y presentes para la predicción. Para el trabajo aquí presentado se ha entrenado y probado un sistema predictor de glucosa en sangre con datos reales recogidos de 10 pacientes diabéticos diferentes, obteniendo predicciones de 30, 60, 90 y 120 minutos con resultados de precisión alentadores.**

***Palabras clave*—Diabetes, Predicción de Glucosa, Fuzzy Rule System**

I. Introducción

La diabetes de tipo I es una enfermedad auto-inmune que provoca la destrucción de las células productoras de insulina (células beta) del páncreas. Un páncreas sano se encarga de regular los niveles de glucosa en sangre mediante la producción de insulina. Gracias a la insulina, las células del organismo pueden absorber la glucosa del torrente sanguíneo. En el caso de un paciente diabético, la ausencia de insulina impide la asimilación de la glucosa, provocando un aumento de los niveles de glucosa en el torrente sanguíneo. La acción del páncreas se sustituye por la inyección manual de insulina en régimen Bolo-Basal. Mantener los niveles de glucosa en sangre dentro del rango recomendado saludable es complejo debido a los numerosos factores que hacen que los niveles de glucosa cambien, a veces de forma repentina.

La motivación de este estudio radica en la necesidad de disponer de modelos fiables de predicción de los niveles de glucosa en sangre para los pacientes con diabetes de tipo I. La predicción correcta de los valores futuros de glucosa ayudará a los pacientes a determinar las dosis de insulina que deben inyectarse para mantener los niveles en el rango saludable y evitar las complicaciones a largo plazo. Para hacer frente a este grado de incertidumbre y complejidad mencionado anteriormente, investigaremos el rendimiento de un sistema basado en reglas difusas de Takagi-Sugeno-Kang (TSK-FRBS) de dos fases. Este tipo de sistemas pueden aportar valores predictivos fiables al problema gracias a su adaptación a situaciones con alta incertidumbre.

La primera fase es un proceso de aprendizaje en el que las funciones de pertenencia y la base de reglas se optimizan mediante un algoritmo genético. En la segunda fase de ajuste, utilizamos un algoritmo genético para realizar la selección y optimización de las reglas.

Para realizar la predicción hemos contemplado dos escenarios que intentan reflejar las situaciones reales a las que se enfrentan los pacientes en tratamiento con insulina: *What-if* y *Agnóstico*. En ambos casos la información de la que disponemos es: los valores recientes de glucosa en sangre medidos por un CGM, así como la ingesta previa de carbohidratos y las inyecciones de insulina. La diferencia entre los dos escenarios radica en si tenemos en cuenta los valores futuros supuestos de las comidas y las inyecciones de insulina (*What-if*) o si sólo consideramos la información pasada y presente para la predicción (*Agnóstico*).

Nuestros resultados experimentales ponen en evidencia que esta técnica produce modelos basados en reglas que permiten una buena predicción de la evolución del nivel de glucosa, y evitan, de forma razonable, ajustarse demasiado a condiciones temporales específicas. Además, dado que la técnica empleada

para el entrenamiento del modelo se caracteriza por ser una optimización de *caja blanca* basada en reglas, el modelo obtenido puede proporcionar información útil incluso para mejorar esos modelos en el futuro.

El resto del trabajo se organiza en las siguientes secciones. La sección II describe, en mayor medida, el problema de la predicción de los niveles futuros de glucosa en sangre en pacientes diabéticos y revisa otras aproximaciones a los problemas presentados recientemente. La sección III explica nuestro enfoque, el flujo de trabajo y las principales técnicas utilizadas en esta investigación, mientras que la sección IV describe la configuración experimental y los resultados de predicción que han obtenido estas técnicas. Las conclusiones y el trabajo futuro se exponen en la sección V.

II. Predicción de Glucosa

Como se ha mencionado en la introducción, predecir los niveles de glucosa es una tarea compleja. Aquellos pacientes diabéticos que son insulinodependientes necesitan estimar cuál será el efecto de la insulina sobre sus niveles de glucosa en sangre, tanto para decidir qué cantidad inyectarse en función de la ingesta de alimentos, como para decidir qué cantidad y qué tipo de alimentos comer, e incluso para tomar decisiones sobre si hacer más o menos ejercicio o con más o menos intensidad. En este contexto, podemos definir el problema de la siguiente manera:

- *Predecir el valor del nivel de glucosa en sangre (BG,de sus siglas en inglés) en un horizonte temporal de H minutos, conociendo la información de los niveles actuales de glucosa en sangre, la ingesta de alimentos (CH), las inyecciones de insulina (I), y la actividad física en el momento de la predicción incluyendo los eventos de las últimas dos horas.*

Se han considerado dos escenarios diferentes (véase también la sección IV-C). El primero, que llamamos *What-if*, nos permite asumir valores prospectivos para la insulina y la ingesta de alimentos. Por lo tanto, los valores registrados en las dos horas inmediatamente posteriores a la hora de predicción de estos registros también pueden utilizarse para entrenar los modelos. En el segundo escenario, denominado *Agnóstico*, buscamos modelos que sólo utilicen la información disponible hasta el momento de la predicción, restringiendo esta información a las dos horas anteriores a ese instante.

Los niveles de glucosa se recogen mediante el uso de sistemas de Monitorización Continua de Glucosa (CGM), muy utilizados hoy en día y que miden los niveles de glucosa cada cinco o quince minutos. Las inyecciones de insulina se registran mediante un dispositivo utilizado para esta tarea, una bomba de insulina automática que lleva el paciente. Las ingestas de alimentos son registradas por el paciente manualmente como estimaciones de la cantidad de gramos de carbohidratos ingeridos.

Hemos estado trabajando en el problema durante la última década y en los últimos años hemos desarrollado un conjunto de metodologías basadas en la Computación Evolutiva. En particular, hemos utilizado dos formas diferentes de abordar el problema:

- Como problemas de predicción de series temporales, uno para cada valor del horizonte temporal.
- Utilizando un modelo mínimo y ajustando los parámetros para cada paciente.

Bajo el primer enfoque obtuvimos muy buenos resultados utilizando los algoritmos de Gramáticas Evolutivas (GE), [1] tanto en la predicción como en la fase de entrenamiento. Bajo el segundo enfoque, en [2] propusimos un método mejorado para la predicción de la tendencia de la glucosa en sangre del paciente basado en el modelo mínimo donde los parámetros se obtienen mediante un algoritmo genético.

Como en muchos otros campos de la medicina, la recogida de datos en pacientes reales es muy compleja, ya que necesitamos un compromiso muy fuerte por parte de los pacientes para recoger los datos correctamente, lo que no siempre es fácil, incluso con la mejor disposición del paciente y la máxima colaboración del equipo médico y sanitario. Por lo tanto, la escasez de cantidades significativas de datos es uno de los principales obstáculos que hemos encontrado a la hora de entrenar los modelos, un problema que se puede observar en la mayoría de los trabajos en esta área: normalmente se considera un número reducido de pacientes, no más de 10. Los modelos de GE entrenados con un conjunto de datos reducido suelen sufrir un sobreajuste y tienen una capacidad de predicción limitada, por lo que es necesario aplicar técnicas adicionales para evitar esta situación. Por ejemplo, en [1] se aplica el aumentado de datos para ampliar el número de registros de datos de glucosa. Podemos encontrar otros trabajos que utilizan datos generados por simuladores, también denominados datos in-silico.

Por supuesto, existen varias aproximaciones basadas en otras técnicas, como las Redes Neuronales Artificiales, e incluso hoy en día se dedica una línea de investigación específica a este problema [3]. En este trabajo analizamos diferentes técnicas de programación genética a la hora de construir modelos de caja blanca para la predicción con horizontes de 30 y 60 minutos. Los resultados se comparan con los métodos clásicos, incluyendo la regresión lineal multivariante, random forest, así como dos tipos de modelos autorregresivo integrado de media móvil.

En este trabajo comparamos el rendimiento de la nueva técnica basada en reglas difusas con los trabajos descritos anteriormente. Nuestra idea es que los modelos basados en reglas pueden ser muy útiles para el paciente y podrían llegar a aplicarse en la gestión diaria de la diabetes.

Los sistemas basados en reglas difusas se han utilizado tradicionalmente como sistemas de control. Se ha demostrado que estos sistemas son capaces de dar muy buenas respuestas a problemas en los que se necesita un cierto control automático. La naturaleza de estos sistemas está basada en reglas y su funcionamiento es similar a cómo el ser humano toma sus decisiones, basándose en reglas con un cierto grado de incertidumbre. El diseño de estos sistemas es complejo, ya que debe hacerse con pleno conocimiento del problema y

tradicionalmente se diseñan ad-hoc. La incorporación de algoritmos genéticos en el diseño ha permitido abordar problemas más complejos, pero con la implicación de una pérdida de interpretabilidad.

Existen diferentes tipos de sistemas basados en reglas difusas (FRBS, sus siglas en inglés), entre los más conocidos están los sistemas Mandani y TSK. Cuando el objetivo es implementar un FRBS para resolver un determinado problema, el tipo de sistema debe ser seleccionado de acuerdo a las características del problema. En el trabajo que aquí se presenta, un FRBS-TSK es el que mejor se adapta a las características del problema, ya que busca una relación matemática entre las variables de entrada y el dato a predecir, en nuestro caso el valor de la glucosa en sangre. En un FRBS-TSK, el consecuente de la regla difusa es una función matemática, que nos permite relacionar matemáticamente los valores de las variables de entrada con el valor a predecir.

III. Metodología

Es interesante recordar que en nuestro conjunto de datos, cada registro individual consiste en un valor de glucosa en sangre a predecir más una ventana histórica con datos de glucosa, carbohidratos e insulina.

A. Módulo FRBS-TSK

El modelo de predicción de glucosa se realiza mediante el algoritmo FRBS-TSK. Este algoritmo establecerá las relaciones entre las variables de entrada del modelo presentado en relación a la variable con los datos a predecir. El modelo predictivo final se generará mediante las conocidas técnicas basadas en reglas difusas utilizando el modelo TSK. La base de conocimientos (KB, de sus siglas en inglés) y la base de datos de reglas (RB, de sus siglas en inglés) necesarias para dicho modelo se optimizarán con un algoritmo genético en dos fases:

- Una fase de aprendizaje del modelo, en la que se optimizan las funciones de pertenencia y las reglas.
- Una fase de ajuste en la que se optimiza el uso de las reglas, así como mejoras en las funciones de pertenencia.

Una vez que el algoritmo se ha optimizado, podemos predecir el valor de la glucosa para los diferentes pacientes. Las variables utilizadas por el FRBS-TSK son:

- Variables de entrada
  - Nivel de glucosa en el momento de la predicción.
  - Valores de glucosa de las dos horas anteriores al momento de la predicción t.
  - Cantidad de hidratos de carbono ingeridos en el momento de la predicción (si los hay).
  - Cantidad de carbohidratos ingeridos en las dos horas anteriores al momento de la predicción.
  - Unidades de insulina administradas en el momento de la predicción (si las hay).
  - Unidades de insulina administradas en las dos horas anteriores al momento de la predicción.
- Variables de salida
  - Valor esperado de la glucosa en H minutos después del momento de la predicción, siendo $H \in \{30, 60, 90, 120\}$ el horizonte de predicción en minutos.

Para el escenario agnóstico, también están disponibles los valores de insulina y carbohidratos durante el intervalo de tiempo $[t, t + H]$.

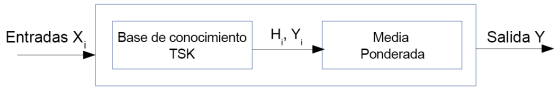


Fig. 1. Modelo FRBS-TSK.

La estructura básica de un sistema TSK consiste en una KB y una RB. Ambas partes del sistema almacenan la base del conocimiento extraído del problema (KB) y las relaciones entre sus variables establecidas por las funciones de pertenencia (MF, de sus siglas en inglés). El conocimiento se representa mediante reglas simples del tipo *IF-THEN*. Las reglas difusas tienen una estructura con un antecedente y un consecuente. El antecedente está compuesto por variables lingüísticas, y el consecuente es una función polinómica de las variables de entrada. Así, el formato de las reglas de un FRBS-TSK tiene la siguiente estructura:

$$\text{Si } X_1 \text{ es } A_1 \ \ldots \ \text{y } X_n \text{ es } A_n$$
$$\text{entonces Y} = p_1 * X_1 + \ \ldots \ + p_n * X_n + p_0$$

donde las variables de entrada del sistema se denotan como X_i, Y es la variable de salida del sistema, p_i representa coeficientes de valores reales y A_i son conjuntos difusos.

Los FRBS-TSK se han aplicado con éxito a un gran número de problemas, particularmente con una alta dimensionalidad de los datos [4]. La novedad de este sistema es el uso de una ecuación de sistema compacta para la estimación del parámetro p_i, utilizando métodos clásicos. Esto da lugar a un sistema preciso que resulta muy útil para la modelización difusa precisa.

El proceso de diseño y optimización del algoritmo METSK-HD se divide en dos fases, *Aprendizaje* y *Ajuste*. En la primera fase de aprendizaje, se genera la BD inicial. Esto se hace mediante el conjunto de reglas TSK de orden cero candidatas para el modelo completo. Con el uso de un Algoritmo Evolutivo Multiobjetivo (MOEA) [5] se realiza este aprendizaje. La segunda fase de ajuste, aplica un post-procesamiento avanzado para el ajuste evolutivo de las MFs basado en dispersión fina. En este proceso, se realiza un ajuste significativo de las reglas difusas. La figura 2 muestra el proceso descrito anteriormente.

B. Fases de optimización del METSK-HD

A continuación se describen las fases en las que se divide el proceso de optimización del METSK-HD.

En una primera fase de *aprendizaje*, se utiliza un MOEA para la KB inicial. Este método se encarga de diseñar un FRBS-TSK inicial para reducir la dimensionalidad y utilizar mecanismos eficaces para garantizar una convergencia rápida

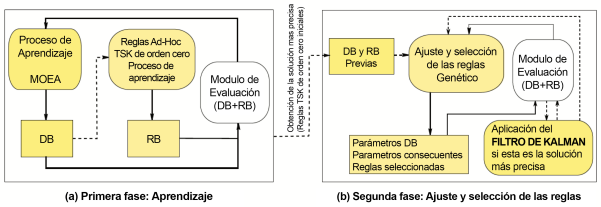


Fig. 2. Proceso de optimización FRBS-TSK.

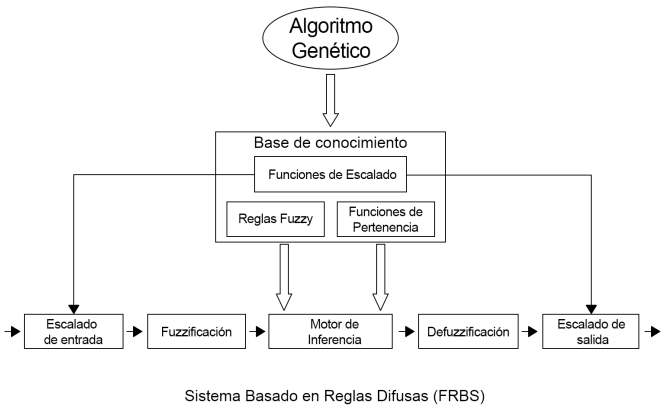


Fig. 3. Procesos implicados en el módulo FRBS.

en conjuntos de datos de regresión de alta dimensión y gran escala. Este proceso se divide en:

Generación de la KB. El algoritmo METSK-HD utiliza el conocido método de Wang y Mendel (WM) [6] para obtener la KB completa. Se ha acotado a un máximo de 100 reglas, que al ser reglas para un sistema TSK, se obtienen con todos los coeficientes con valor 0 y los términos independientes calculados como la media de los ejemplos cubiertos por la regla ponderados por su coincidencia.

Codificación de la BD. Se considera un sistema de representación lingüística de 2 tuplas. Esto nos permite utilizar un único parámetro en el proceso de optimización, el desplazamiento lateral de las MFs, a diferencia de los sistemas anteriores que requieren el uso de los tres parámetros que describen la forma triangular de las funciones de pertenencia utilizadas. En este enfoque, se utiliza el mismo desplazamiento $[-0.1, 0.1]$ para todas las variables.

Objetivos: los objetivos en el proceso de optimización son:

- minimizar el error cuadrático medio.
- minimizar el número de reglas.
- maximizar el grado de cobertura de los ejemplos.

Para ello, partimos de una población inicial en la que cada cromosoma contiene el mismo número de etiquetas lingüísticas para todas las variables del sistema (mínimo 2 y máximo 7), junto con un desplazamiento lateral inicial calculado aleatoriamente. El operador de cruce utilizado es Parent Centric BLX (PCBLX) [7] y el operador de mutación disminuirá la granularidad en un gen seleccionado al azar. Por último, se ha incluido un mecanismo de prevención del incesto en las partes que siguen los conceptos del CHC [8].

Segunda fase de *ajuste*. Una vez que se obtiene una KB completa de orden cero en la primera fase, se aplica un MOEA de posprocesamiento para realizar el ajuste de MF y la selección de reglas, lo que ayudará a mejorar significativamente la precisión. El esquema de codificación C incluye:

$C = C_t + C_s + C_c$

donde:

- Esquema clásico de funciones de pertenencia representadas por un triángulo y ajustados en la segunda fase. C_t
- Codificación de la selección de reglas (1 variable asignada 0 no asignada). C_s
- Codificación de los consecuentes de las reglas TSK. C_c

El control de incesto se aplica a la parte del cromosoma C_t, el cruce BLX-0.5 a C_c. La mutación se aplicará a C_s.

Para estimar los coeficientes de la función consecuente polinómica en las reglas TSK, se utiliza la técnica clásica del filtro de Kalman [9]. Por último, el algoritmo de optimización tiene un operador de reinicio que se aplica cuando un valor de umbral L es inferior a cero $L < 0$. El operador de reinicio se aplica copiando sólo el mejor individuo para cada uno de los tres objetivos como los tres primeros individuos de una nueva población inicial.

IV. Resultados Experimentales

A. Configuración Experimental

Con el objetivo de desarrollar y ajustar el mejor modelo posible, hemos seguido la metodología clásica que divide el conjunto de datos en dos subconjuntos: de entrenamiento y de prueba (70% y 30% respectivamente). Todos los modelos se han entrenado y ajustado en el subconjunto de entrenamiento y, finalmente, hemos decidido qué modelo se va a probar en el segundo subconjunto. Para ello hemos considerado tres opciones:

- El modelo con mejor rendimiento durante la fase de entrenamiento.
- Un conjunto de los mejores modelos durante la primera fase. La predicción final en el subconjunto de prueba es la media de las predicciones de todos los modelos del conjunto.
- Se consideró una tercera opción en la que se extrajo un 20% del subconjunto de entrenamiento para una validación previa. Una vez entrenados los modelos en el (80%) restante del subconjunto de entrenamiento, el modelo con mejor rendimiento en ese 20% fue el seleccionado para ser probado.

Además, hemos comparado nuestros resultados con otros trabajos previos que dividieron los datos en subconjuntos de entrenamiento y de prueba.

B. Datos

Es importante destacar que hemos utilizado un conjunto de datos propios, recogidos por nuestro equipo gracias a la

TABLE I
RESULTADOS CON EL RMSE EN LAS PREDICCIONES DE LOS MODELOS

	What-if				**Agnóstico**			
	t+30	**t+60**	**t+90**	**t+120**	**t+30**	**t+60**	**t+90**	**t+120**
	Predicción con el Mejor modelo en la fase de entrenamiento							
Media	30.0309	42.1644	53.3563	56.9629	29.0028	42.5097	53.2706	64.1276
Desv. típica	7.8433	9.1570	12.2502	14.8179	8.6621	11.2201	14.5181	28.8038
	Predicción con el Mejor modelo en la fase de validación							
Media	23,8262	36,6530	44,2958	48,0247	24,1106	37,5778	45,9800	51,7979
Desv. típica	3,8952	5,7243	7,9124	8,9460	4,1754	5,8021	7,6968	10,2609
	Predicción con el Mejor modelo Ensemble							
Media	23,3201	35,8437	42,8433	46,6731	23,4763	36,6025	44,7387	49,3781
Desv. típica	3,7438	5,2864	6,9592	8,1230	4,0147	5,5616	6,6689	8,1696
	Predicción con el Mejor modelo en la fase de test							
Media	18.8757	31.1822	36.9369	39.7829	19.0472	31.7357	38.9818	42.6586
Desv. típica	2.5728	5.5997	7.5413	8.5451	2.6953	5.0279	7.1544	8.5664

TABLE II
CLASIFICACIÓN GENERAL DEL BGLP CHALLENGE.

		30 minutos		60 minutos		
	Trabajo	RMSE	MAE	RMSE	MAE	Total
	Harry Rubin-Falcone [10]	18.22	12.83	31.66	23.60	86.31
	Hadia Hameed [11]	19.21	13.08	31.77	23.09	87.15
	Taiyu Zhu [12]	18.34	13.37	32.21	24.20	88.12
	Tao Yang [13]	19.05	13.50	32.03	23.83	88.41
	Robert Bevan [14]	18.23	14.37	31.10	25.75	89.45
	Xiaoyu Sun [15]	19.37	13.76	32.59	24.64	90.36
	David Joedicke [16]	19.60	14.25	34.12	25.99	93.96
	Ning Ma [17]	20.03	14.52	34.89	26.41	95.85
Agnóstico	Mejor modelo en fase de entrenamiento	29.00	21.01	42.51	32.02	124.54
	Ensemble Model	23.48	17.26	36.60	27.89	105.25
	Mejor modelo en fase de validación	24.11	17.54	37.58	28.47	107.70
	Mejor modelo en fase de test	19.05	14.40	31.74	24.81	90.00
What-if	Mejor modelo en fase de entrenamiento	30.03	21.22	42.16	32.13	125.54
	Ensemble Model	23.32	17.20	35.84	27.43	103.79
	Mejor modelo en fase de validación	23.83	17.39	36.65	27.87	105.74
	Mejor modelo en fase de test	18.88	14.31	31.18	24.64	89.01

colaboración de los pacientes del Hospital Príncipe de Asturias de Alcalá de Henares, España, y de su personal médico. El comité ético del hospital aprobó la recogida de datos para nuestros estudios y todos los pacientes firmaron previamente un consentimiento.

La obtención de estos datos no es una tarea sencilla debido a diferentes razones. En primer lugar, se trata de información sensible, y luego los pacientes y el personal médico tienen que implicarse mucho en esta tarea. Los pacientes tienen que llevar dispositivos, tomar notas y registrar la información durante varios días y con las mínimas molestias posibles. Es habitual descartar parte de los datos debido a errores del paciente o a fallos de los dispositivos electrónicos. Los estudios en este campo suelen tratar con conjuntos de datos pequeños, con unos diez pacientes o incluso menos.

En este trabajo se han utilizado los datos de diez pacientes con Diabetes Mellitus Tipo 1 (DMT1). Los datos se han obtenido en intervalos de cinco minutos utilizando el sistema de monitorización continua de glucosa (CGM) de Medtronic durante al menos diez días completos. Las mediciones no fueron estrictamente llevadas a cabo en días consecutivos ni durante el mismo número de días por paciente. La media de días con datos es de 44,80 (+/- 30,73). Las características de los pacientes son mujeres (80 %), edad media 42,30 (+/- 11,07) años, peso 64,78 (+/- 13,31) kg, años de enfermedad 27,20 (+/- 10,32), años con terapia de bomba 10 (+/- 4,98), HbA1c media de 7,27 % (+/- 0,5 %).

C. *Escenarios What-if y Agnóstico*

Para la predicción de la glucosa en sangre solemos definir dos escenarios diferentes *What-if* y *Agnóstico*. La principal diferencia entre ellos es la información utilizada para construir los modelos. En el escenario *What-if* podemos incluir eventos futuros para algunas de las variables de entrada. En particular, incluimos los carbohidratos y las dosis de insulina que se administrarán desde el momento actual hasta el momento del horizonte de predicción. Por otro lado el escenario *Agnóstico* sólo se tienen en cuenta los valores de las variables de entrada que son anteriores al instante de tiempo inmediatamente posterior a la petición de predicción.

Hemos realizado ocho tipos de experimentos probando los dos escenarios diferentes explicados (*What-if* y *Agnóstico*) y

cuatro horizontes temporales diferentes para la predicción (30, 60, 90 y 120 minutos).

D. Resultados

En la tabla I podemos encontrar los resultados obtenidos en el subconjunto de prueba de las diferentes opciones explicadas en la sección IV-A. La tabla muestra el RMSE para los dos escenarios y para los cuatro horizontes temporales. Cada fila de la tabla indica la media y la desviación estándar de todas las ejecuciones de los 10 pacientes.

Como podemos observar en la tabla I, la elección del mejor modelo durante la fase de validación mejora los resultados obtenidos al utilizar el mejor modelo durante la fase de entrenamiento. Y a su vez, las predicciones del mejor modelo Ensemble mejoran ligeramente las cifras obtenidas por el primero. Esta situación no cambia para los cuatro horizontes de predicción y los dos escenarios presentados en este artículo. También hemos incluido los resultados de los mejores modelos obtenidos de la ejecución de todos los modelos en la fase de test. Obviamente, esta no es una opción realista, pero nos da una idea del margen de mejora que todavía puede ofrecer esta técnica. Finalmente, en la tabla II mostramos una comparación de nuestra técnica en contraposición a los mejores modelos registrados en el Blood Glucose Level Prediction (BGLP) Challenge [18]. Aunque se trata de un trabajo preliminar que aún necesita un significativo trabajo de ajuste, los resultados obtenidos pueden considerarse competitivos frente a los de otras propuestas, invitando a seguir trabajando. Es importante destacar que el conjunto de datos utilizado en el BGLP Challenge está compuesto por 6 pacientes con características totalmente diferentes a los 10 pacientes utilizados en este trabajo.

V. Conclusiones

En este trabajo hemos investigado el rendimiento de un sistema basado en reglas difusas en problemas de predicción del nivel de glucosa. Hasta donde sabemos, es la primera vez que se aplica este tipo de enfoque a este campo. La principal ventaja es que los modelos obtenidos son interpretables (modelos de caja blanca), ya que están compuestos por un conjunto de unas diez reglas difusas. Los resultados experimentales muestran que esos modelos son competitivos en comparación con otros enfoques de caja negra como los basados en redes neuronales.

En el trabajo presentado aquí, se han utilizado los datos de 10 pacientes con diferentes características. En el futuro, gracias a las colaboraciones con el hospital Ramón y Cajal de Madrid, podremos utilizar una base de pacientes más significativa.

Agradecimientos

Comunidad de Madrid y de los fondos FEDER bajo la proyecto Y2018/NMT-4668 (Micro-Stress- MAP-CM) y FEDER B2017/BMD3773 (GenObIA-CM). Ministerio de Economía y Competitividad bajo el proyecto TIN2017-85727-C4-{2,4}-P, RTI2018-095180-B-I00 y PID2020-115570GB-C21. Junta de Extremadura, Consejería de Comercio y Economía, del Fondo Europeo de Desarrollo Regional, una manera de construir Europa, bajo el proyecto IB16035 proyecto GR15068 y GR18049. Fundación Eugenio Rodriguez Pascual 2019.

Referencias

[1] Jose Velasco, Oscar Garnica, Juan Lanchares, Marta Botella, and Ignacio Hidalgo. Combining data augmentation, edas and grammatical evolution for blood glucose forecasting. *Memetic Computing*, 10, 06 2018.

[2] Carlos Cervigon, J. Ignacio Hidalgo, Marta Botella, and Rafael-J Villanueva. A genetic algorithm approach to customizing a glucose model based on usual therapeutic parameters. *Progress in Artificial Intelligence*, 6(3):255–261, 2017.

[3] Kerstin Bach, Razvan C. Bunescu, Cindy Marling, and Nirmalie Wiratunga, editors. *Proceedings of the 5th International Workshop on Knowledge Discovery in Healthcare Data co-located with 24th European Conference on Artificial Intelligence, KDH@ECAI 2020, Santiago de Compostela, Spain & Virtually, August 29-30, 2020*, volume 2675 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2020.

[4] María José Gacto, Marta Galende, Rafael Alcalá, and Francisco Herrera. Metsk-hde: A multiobjective evolutionary algorithm to learn accurate tsk-fuzzy systems in high-dimensional and large-scale regression problems. *Information Sciences*, 276:63–79, 2014.

[5] Kalyanmoy Deb. Multi-objective optimisation using evolutionary algorithms: an introduction. In *Multi-objective evolutionary optimisation for product design and manufacturing*, pages 3–34. Springer, 2011.

[6] L-X Wang and Jerry M Mendel. Generating fuzzy rules by learning from examples. *IEEE Transactions on systems, man, and cybernetics*, 22(6):1414–1427, 1992.

[7] Manuel Lozano, Francisco Herrera, Natalio Krasnogor, and Daniel Molina. Real-coded memetic algorithms with crossover hill-climbing. *Evolutionary computation*, 12(3):273–302, 2004.

[8] Larry J Eshelman. The chc adaptive search algorithm: How to have safe search when engaging in nontraditional genetic recombination. In *Foundations of genetic algorithms*, volume 1, pages 265–283. Elsevier, 1991.

[9] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. 1960.

[10] Harry Rubin-Falcone, Ian Fox, and Jenna Wiens. Deep residual time-series forecasting: Application to blood glucose prediction. In *Proceedings of the 5th International Workshop on Knowledge Discovery in Healthcare Data co-located with 24th European Conference on Artificial Intelligence, KDH@ ECAI*, pages 29–30, 2020.

[11] Hadia Hameed and Samantha Kleinberg. Investigating potentials and pitfalls of knowledge distillation across datasets for blood glucose forecasting. In *Proceedings of the 5th Annual Workshop on Knowledge Discovery in Healthcare Data*, 2020.

[12] Taiyu Zhu, Xi Yao, Kezhi Li, Pau Herrero, and Pantelis Georgiou. Blood glucose prediction for type 1 diabetes using generative adversarial networks. In *CEUR Workshop Proceedings*, volume 2675, pages 90–94, 2020.

[13] Tao Yang, Ruikun Wu, Rui Tao, Shuang Wen, Ning Ma, Yuhang Zhao, Xia Yu, and Hongru Li. Multi-scale long short-term memory network with multi-lag structure for blood glucose prediction.

[14] Robert Bevan and Frans Coenen. Experiments in non-personalized future blood glucose level prediction. In *CEUR Workshop Proceedings*, volume 2675, pages 100–104, 2020.

[15] Xiaoyu Sun, Mudassir Rashid, Mert Sevil, Nicole Hobbs, Rachel Brandt, Mohammad Reza Askari, Andrew Shahidehpour, and Ali Cinar. Prediction of blood glucose levels for people with type 1 diabetes using latent-variable-based model.

[16] David Joedicke, Oscar Garnica, Gabriel Kronberger, J Manuel Colmenar, Stephan Winkler, J Manuel Velasco, Sergio Contador, and J Ignacio Hidalgo. Analysis of the performance of genetic programming on the blood glucose level prediction challenge 2020.

[17] Ning Ma, Yuhang Zhao, Shuang Wen, Tao Yang, Ruikun Wu, Rui Tao, Xia Yu, and Hongru Li. Online blood glucose prediction using autoregressive moving average model with residual compensation network.

[18] 2020 The 5th International Workshop on Knowledge Discovery in Healthcare Data. 24th European conference in Artificial Intelligence (ECAI). Blood glucose level prediction challenge, 2020.

Modelo híbrido basado en aprendizaje profundo de creación y etiquetado de células escamosas procedentes de citologías cervicovaginales

Andrés Bueno-Crespo
Escuela Politécnica
Universidad Católica de Murcia
Murcia, España
abueno@ucam.edu

Ana Ortiz-González
Dept. de Anatomía Patología
Hospital General Univ. Santa Lucía
Cartagena, España
anaortizg91@gmail.com

Raquel Martínez-España
Escuela Politécnica
Universidad Católica de Murcia
Murcia, España
rmartinez@ucam.edu

José Martínez-Más
Dept. de Obstetricia y Ginecología
Centro CIAGO
Murcia, España
jomartmas@gmail.com

Miguel Ángel Gragera
Escuela Politécnica
Universidad Católica de Murcia
Murcia, España
magragera@alu.ucam.edu

Manuel Remezal-Solano
Dept. de Obstetricia y Ginecología
Hospital Clínico Univ. Virgen de la Arrixaca
Murcia, España
mremezals@insermed.com

Juan-Pedro Martínez-Cendán
Dept. de Obstetricia y Ginecología
Hospital General Univ. Santa Lucía
Cartagena, España
jmartinezcen@gmail.com

Sebastián Ortíz-Reina
Dept. de Anatomía Patología
Hospital General Univ. Santa Lucía
Cartagena, España
sortizreina@yahoo.es

Andrés Muñoz
Escuela Politécnica
Universidad Católica de Murcia
Murcia, España
amunoz@ucam.edu

***Abstract*—El cáncer de cuello uterino es uno de los más agresivos y frecuentes si no se detecta y trata en fases iniciales. La detección de esta enfermedad se realiza a través de cribados que son analizados a nivel celular manualmente por patólogos expertos. El aprendizaje profundo permite crear modelos de clasificación automática para facilitar la labor de detección. El problema de los modelos de aprendizaje profundo es la necesidad de disponer de un volumen considerable de datos , pero en el caso de diagnóstico médico, no siempre se dispone de tal cantidad de datos. Para paliar este problema, en este estudio se propone un nuevo enfoque computacional para crear y etiquetar células escamosas de cérvix. Este enfoque está basado en aprendizaje profundo y se basa en una combinación de la técnicas de redes neuronales antagónicas y de redes neuronales profundas convolucionales. Los resultados obtenidos son satisfactorios y plantean la creación de una nueva base de datos sintética con imágenes creadas y etiquetadas de forma automática.**

***Index Terms*—Redes generativas antagónicas, Redes neuronales convolucionales, Cáncer uterino, Clasificación de células, Aprendizaje Profundo**

I. Introducción

El cáncer de cuello uterino es uno de los cánceres más frecuentes, siendo el sexto más detectado en Europa y el décimo a nivel mundial. En los países desarrollados existen programas de cribado que permiten detectar esta enfermedad a edad temprana [2]. Sin embargo, en países en desarrollo esta enfermedad supone un 83% de los casos de cáncer en mujeres, acabando en muerte un 90% de estos casos debido a la falta de diagnóstico precoz. Esta enfermedad comienza en las células del cuello del útero. La técnica de cribado es un proceso subjetivo realizado por patólogos expertos donde examinan detenidamente cada muestra de estas células. Aunque el proceso es fiable, hay que tener en cuenta que produce un número de falsos positivos y falsos negativos, dependiendo de la calidad de la muestra, errores en el procesamiento o la fatiga del experto, entre otras situaciones.

La técnica de cribado más destacada es la citología cervical, propuesta por Papanicolaou en 1928. Esta técnica de análisis celular se centra en los cambios celulares tempranos que podrían conducir al cáncer de cuello de útero y demostró ser la mejor prueba de cribado para el cáncer cervical y sus cambios celulares tempranos [6]. Después de esta técnica se han desarrollado modificaciones para facilitarla como frotis líquidos, tinción y procesos automatizados, pero para cualquier técnica, el patólogo experto tiene que seguir examinando la muestra de forma manual. La sensibilidad de los cribados realizados por patólogos de alto nivel va del 68,8% al 93,8% (media del 77,4%) y la especificidad del 70,6% al 95,6% (media del 81,3%), siendo la media de precisión del 79,4% [12], [13].

Respecto a la clasificación de las células, existe un consenso para hacerla en cuatro categorías [7], [8]: (1) células sin lesión (células normales o benignas), (2) células con lesión intraepitelial escamosa de bajo grado (L-SIL), (3) células con lesión intraepitelial escamosa de alto grado (H-SIL) y (4) células escamosas atípicas de significado indeterminado (ASC-

US). Estas últimas indican que las anormalidades celulares son más marcadas que los cambios reactivos pero no tanto como para ser clasificadas como lesión intraepitelial.

En un estudio previo [11] se presentó un modelo de aprendizaje profundo (Deep Learning, DL) para mejorar la especificidad, sensibilidad y precisión en el proceso de la clasificación automática de células escamosas. Una de las principales dificultades encontradas fue la obtención de una cantidad suficiente de células escamosas reales para obtener un modelo robusto y con una precisión mayor o igual a la de un experto. Ante esta dificultad, en dicho estudio se propuso un método de mezclado de imágenes para ampliar el conjunto de células de entrenamiento. Se obtuvo un conjunto de imágenes suficiente para obtener un modelo de clasificación de imágenes de células escamosas que igualaba la precisión humana. Sin embargo, la obtención de células reales siguen siendo un problema importante, debido por un lado a la burocracia necesaria para obtener imágenes sanitarias y por otro lado en la necesidad de que el patólogo experto deba de ir etiquetando cada célula manualmente de una en una, lo que provoca una lentitud considerable. Además, también existe la problemática de encontrar células de todas las categorías en cantidad equilibrada, ya que no en todas las citologías pueden encontrarse muestras de todas ellas.

En este trabajo presentamos un nuevo enfoque computacional para crear y etiquetar nuevas células escamosas de las cuatro categorías definidas previamente mediante un modelo híbrido basado en aprendizaje profundo. Siendo el principal objetivo y aportación de este trabajo la innovación en el ámbito de detección de imágenes de células cancerígenas mediante citologías cervicovaginales. Para ello, se presenta un modelo híbrido compuesto de la fusión de una red neuronal antagónica, conocida como GAN, junto con una red neuronal de aprendizaje profundo convolucional (DCNN). El objetivo del modelo es crear nuevas células escamosas artificiales, partiendo de células reales que permitan ampliar las bases de datos de las diferentes categorías de células para poder entrenar modelos de clasificación, con resultados más precisos que los conseguido por los patólogos expertos. Así, los errores, falsos negativos y falsos positivos que tienen las técnicas de diagnóstico actuales serían mejorados. Los objetivos específicos del trabajo son los de crear imágenes de células nuevas partiendo de células reales no preprocesadas. Esto implica que las imágenes de las células creadas contienen diferentes elementos de fondo que se encuentran en la muestra, así como células superpuestas con otras plegadas o con artefactos de tinción debido a la técnica de recogida de muestras y su procesamiento.

Este artículo está organizado como sigue: La sección II revisa algunos sistemas y técnicas utilizadas para la clasificación de imágenes y la utilización de las redes antagónicas para la generación de nuevos datos . En la sección III se describe el modelo híbrido propuesto. En la sección IV se describen los datos y los parámetros de los experimentos llevados a cabo para la evaluación del modelo. A continuación, en la sección V, se presentan y discuten los resultados obtenidos. Finalmente, en la sección VI, se presentan las conclusiones y el trabajo futuro

II. Estado del Arte

El campo del Machine Learning y Deep Learning ha sido ampliamente utilizado para la investigación en el ámbito de la salud, como por ejemplo, la creación de modelos de Machine Learning mediante el análisis de una gran cantidad de datos para predecir enfermedades [1]. Pero los avances más importantes y que tienen gran relación con el trabajo que se va a desarrollar, son los análisis de imágenes mediante técnicas de Deep Learning, concretamente usando DCNN para clasificación de imágenes.

Uno de los artículos publicados más importantes en relación con las DCNN es el publicado en la conferencia internacional IEEE para el procesamiento de imágenes y versa sobre la detección de tuberculosis en imágenes de radiografías [4]. También podemos ver usos más contemporáneos de gran utilidad, como por ejemplo el análisis de radiografías para clasificar la gravedad de la enfermedad COVID-19 usándose redes neuronales, donde se logró alcanzar un 93.1% de efectividad en la clasificación [3]. Otra aplicación de DCNN para clasificar cáncer de mama se presenta en [17]. En este estudio, sus autores desarrollan un modelo de DCNN muy preciso y fiable con el objetivo de clasificar tipos y subtipos de cáncer de mama utilizando imágenes histopatológicas teñidas con hematoxilina-eosina y preparaciones inmunohistoquímicas. La clasificación de imágenes mediante datos de espectrometría de masa utilizando DCNN se detalla en [18]. Los autores de este trabajo obtienen resultados muy interesantes proponiendo una arquitectura adaptada a su problema, basada en DCNN, definiendo una estrategia para interpretar el análisis de los modelos. En [19] se plantea un modelo de DCNN para clasificar células del cérvix uterino. En este caso, los autores realizan un procesamiento previo de las imágenes limpiando y eliminando cualquier artefacto que pudiera dificultar el proceso de clasificación en células normales y atípicas. También realizaron en este trabajo distintas sesiones de preentrenamiento de las redes neuronales con distintas imágenes preprocesadas. Para evaluar su trabajo, utilizan una base de datos pública y obtienen resultados interesantes mediante la extracción de características de las imágenes. Es de destacar que en este trabajo, todas las células utilizadas tanto en el entrenamiento como en la evaluación se presentan como células aisladas y libres de artefactos y solapamientos. Esta no es la situación habitual en la práctica clínica real cuando se examina una citología cervicovaginal, puesto que, como se describe más adelante, estas muestras habitualmente se presentan junto a una gran diversidad de estirpes celulares, así como agentes bacterianos saprófitos o patógenos, agregaciones celulares, solapamientos y otros artefactos propios del procesamiento de la muestra.

Sin embargo, el problema de clasificación de imágenes médicas mediante la tecnología Deep Learning concluyen en muchos casos con la falta de conjuntos de datos con mayor cantidad de imágenes. De ahí, el reciente interés en el uso de

redes GAN (redes neuronales generativa antagónicas, también conocidas como adversarias). Estas redes tienen la capacidad de imitar distribuciones de datos y sintetizar imágenes con niveles de realismo sin precedentes. Esto ha provocado nuevas vías para salvar la brecha entre el aprendizaje supervisado y la generación de imágenes [14]. Los autores de [5] proponen la utilización de las redes GAN's para la creación de forma artificial de imágenes de resonancias magnéticas. En [16], los autores proponen la creación de imágenes cerebrales de PET para diferencias y poder clasificar tres etapas diferentes de la enfermedad de Alzheimer: control normal, deterioro cognitivo leve y enfermedad de Alzheimer. En el estudio [15] se propone el uso de una GAN para crear imágenes sintéticas con el objetivo de aumentar un conjunto de datos sobre lesiones hepáticas para poder obtener una clasificación de imágenes de forma automática con una mayor sensibilidad y especificidad.

III. Modelo híbrido de etiquetado de células

El sistema híbrido propuesto utiliza una red generativa adversaria y una DCNN en su salida, que nos va a permitir eliminar imágenes que no cumplen con ciertos criterios diagnósticos. Las redes generativas antagónicas son un sistema de aprendizaje no supervisado donde dos redes compiten entre si con un objetivo común. Para el modelo GAN, se ha utilizado una red StyleGAN2, la cual es una red generativa adversaria que se basa en StyleGAN (utilizada para la generación de imágenes, principalmente de rostros y desarrollada por NVIDIA [9]) con varias mejoras. En primer lugar, se rediseña la normalización adaptativa de las instancias y se sustituye por una técnica de normalización denominada demodulación de pesos. En segundo lugar, se introduce un esquema de entrenamiento mejorado sobre el crecimiento progresivo, que logra el mismo objetivo, el entrenamiento comienza centrándose en imágenes de baja resolución y luego cambia progresivamente el enfoque a resoluciones cada vez mayores, sin cambiar la topología de la red durante el entrenamiento. Además, se proponen nuevos tipos de regularización como la regularización perezosa y la regularización de la longitud de la trayectoria [10]. Finalmente, a salida de la GAN, el modelo propuesto utiliza una DCNN entrenada previamente para discriminar entre los diferentes tipos de diagnósticos clínicos. Esto es necesario, ya que se entrenan una GAN por cada modalidad diagnóstica y las diferencias entre ellas, en algunos casos son muy pequeñas. Cabe destacar que la GAN aprende una tipología de célula para un determinado diagnóstico y por lo tanto no dispone de información sobre los otros diagnósticos.

La red StyleGan2 nos permite generar células completamente nuevas con las características de las imágenes de entrada. Esto se consigue mediante la transformación de las imágenes del espacio latente en la red generativa, este espacio podría definirse como un lugar de n-dimensiones donde se representan las características que ha aprendido la red generativa antes de que vuelva a reconstruir la imagen. Respecto a la DCNN, esta red se encuentra entrenada para clasificar y etiquetar células escamosas de las cuatro categorías diagnósticas mencionadas, L-SIL, H-SIL, ASC-US y Benignas. La ventaja de fusionar ambas técnicas es que la red StyleGan2 no conoce inicialmente la categoría de célula que tiene que generar de forma sintética, es con la DCNN cuando esta célula es etiquetada. Además al disponer de las 4 categorías diagnósticas, los errores de las células mal etiquetadas pueden ser estudiados para ajustar los parámetros de la StyleGan2 para que detecte o tenga una granularidad más fina en determinados aspectos que pueden ser similares entre categorías.

IV. Experimentos

En esta sección se plantean los experimentos para evaluar el modelo híbrido descrito. Primeramente se presenta el conjunto de datos inicial para la generación artificial de células escamosas. Después se discutirá sobre el ajuste de parámetros realizado para el modelo propuesto y por último se mostrarán los resultados del modelo.

A. Conjunto de datos inicial

Partimos de un conjunto de imágenes reales de células escamosas. La muestra de estas células se tomó mediante un frotis cervical utilizando el sistema BD SurePath® de Papanicolaou de base líquida y se procesó mediante el sistema automatizado y robotizado PrepStain®. iScan Coreo® (un escáner de campo iluminado para portaobjetos), fue el sistema de escaneo utilizado en aumento ×20. La Figura 2 muestra una parte del resultado de la imagen inicial con todas las células obtenida por el scaner. La muestra con todas las células se divide en imágenes de tamaño 200 × 200 píxeles donde cada imagen contiene una célula escamosa en alta resolución. Estas células fueron clasificadas por patólogos expertos según la categorización explicada anteriormente, propuesta por el Consenso de Bethesda 2014 [7].

Es importante mencionar que el conjunto de imágenes reales se obtuvo mediante un protocolo aprobado por el Comité de Ética de la Universidad Católica de Murcia, código CE012005. Los resultados se comunicaron siguiendo las directrices STARD (Standards for Reporting of Diagnostic Accuracy) para estudios de precisión diagnóstica. En total el conjunto de datos inicial constaba de las siguientes imágenes:

- 450 imágenes de células de categoría Benigna.
- 213 imágenes de células de categoría L-SIL
- 323 imágenes de células de categoría ASC-US
- 419 imágenes de células de categoría H-SIL

Con este conjunto de datos reales y utilizando el modelo híbrido presentado, se han creado un total de 8000 células artificiales, siendo 2000 de cada categoría. Entre estas 8000 células iniciales, en la parte convolucional del modelo, serán seleccionadas aquellas que sean clasificadas correctamente dentro de su categoría correspondiente.

B. Parámetros óptimos para el modelo propuesto

Para la optimización de los parámetros del modelo propuesto, compuesto por varias redes StyleGan2 [10] y una red DCNN, se han testeado varias configuraciones y redes para el diseño de la red DCNN, mientras que para la red StyleGan2, no se han hecho modificaciones respecto al modelo original.

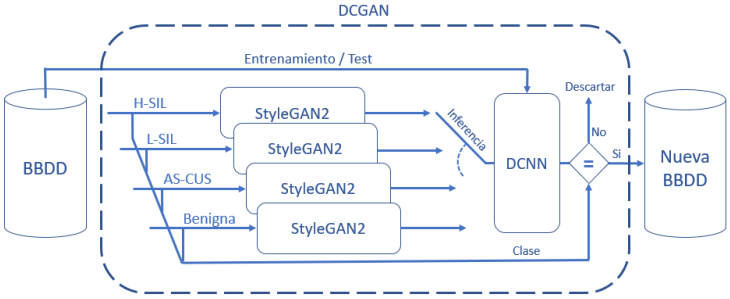


Fig. 1: Modelo híbrido propuesto donde se combinan una GAN y una DCNN para la generación de imágenes de células.

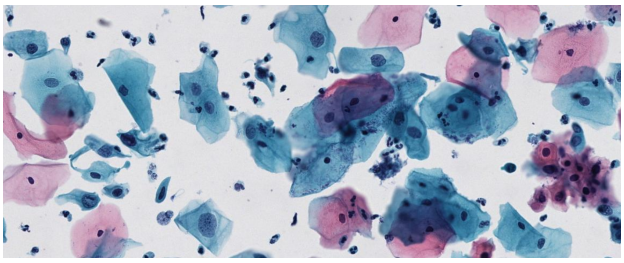


Fig. 2: Recorte de una imagen microscópica de una muestra procedente de una citología de las células escamosas mediante escáner.

Parámetros	Valor
Congelación de capas	30 iniciales
Optimizador	Adam
Tasa de aprendizaje	0.001
Política tasa aprendizaje	Step
Reducción tasa aprendizaje	0.003
Momento	0.9
Función de pérdida	categorical_crossentropy
Métrica	accuracy
Épocas	100
Parada	EarlyStopping
Modo	GPU

TABLE I: Parámetros usados para la red convolucional que forma parte del modelo híbrido propuesto

La Figura 3 muestra los cinco modelos de DCNN testeados y su porcentaje de acierto en predicción sobre las cuatro categorías diagnósticas. Finalmente, se ha optado por una red Xception [20], donde se ha conseguido una precisión de clasificación para las cuatro clases diagnósticas del 75.8% y un 95.8% en la clasificación de benignas respecto a las demás con categorías que deben ser revisadas por el patólogo experto (H-SIL, L-SIL y AS-CUS). La Tabla I muestra el ajuste realizado para la optimización de los parámetros de la red Xception donde previamente se le ha hecho un aprendizaje por transferencia utilizando la base de datos de imágenes ImageNet [21].

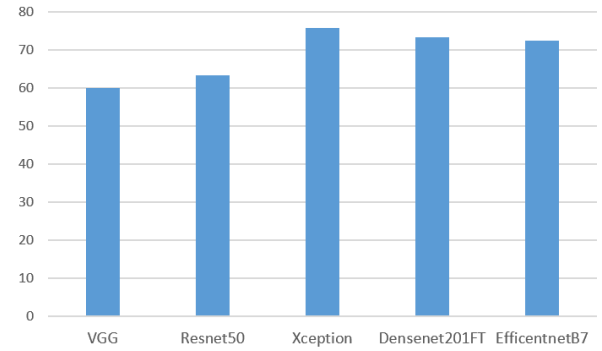


Fig. 3: Porcentaje de acierto en predicción de diferentes modelos de redes DCNN teniendo en cuenta la división del conjunto de datos en cuatro categorías diagnósticas.

V. Resultados y Discusión

Tras validar y optimizar los diferentes parámetros, el modelo propuesto creó 8000 imágenes sintéticas que fueron clasificadas según la matriz de confusión mostrada en la Tabla II. En esta tabla se puede visualizar cómo la parte del modelo centrado en la DCNN, al estar entrenado de forma categórica para las cuatro categorías, consigue clasificar en más de un 75% de media todas las células con precisión. Analizando en detalle tenemos que para las células benignas, el modelo crea y clasifica correctamente 1773 células de las 2000 propuestas. Los fallos los tiene con 143 imágenes con las células ASC-US, 6 imágenes con células H-SIL y 78 imágenes con células L-SIL. Hay que destacar que las células ASC-US tras un análisis mas profundo con otras pruebas diagnósticas, los patólogos pueden determinar que la células es dañina (L-SIL, H-SIL) o que es benigna. Esto viene determinado por la poca diferencia a nivel visual de las células benignas con respecto a las ASC-US en algunas situaciones. Respecto a las células H-SIL, de las 2000 creadas sintéticamente, 1752 son creadas y clasificadas de forma correcta. El resto de células son clasificadas como ASC-US (127) y como L-SIL (102). Hay que destacar que solamente 19 de las 2000 han sido clasificadas como benignas, por lo que el modelo diferencia y crea imágenes de células H-SIL con bastante precisión y cuando se equivoca lo hace para tipos de células en su mayoría que conllevan un revisión adicional con otro tipo de pruebas en las pacientes. Las imágenes de la categoría ASC-US etiquetadas correctamente son 1426 de las 2000. De nuevo al igual que ocurría con las H-SIL, las células ASC-US tienen

un acierto bastante satisfactorio, ya que cuando se equivocan lo hacen clasificando las células ASC-US como de L-SIL (427 imágenes) y como H-SIL (102 imágenes), solamente tienen una mala clasificación 45 de las 2000 que son clasificadas como benignas. Por último Las células L-SIL son las que peor resultados consiguen comparadas con el resto de categorías. En concreto las células L-SIL correctamente etiquetadas son 1138 de las 2000, desviándose el error con 173 células L-SIL consideradas como H-SIL y 585 consideradas como ASC-US. En estos casos aunque el etiquetado es erróneo, el error producido no sería grave ya que se encontraría dentro del conjunto de categorías con necesidad de más pruebas diagnósticas. El error a considerar serían las 104 células etiquetadas como benignas. Aquí, hay que considerar que las células L-SIL son las que contienen la enfermedad en edad muy temprana, por lo tanto, la diferencia visual entre las benignas y las L-SIL puede no ser apreciable si no se detallan características más profundas. Además el mayor grado de error en etiquetado lo tienen estás células porque son de la categoría que menor número tiene de células reales por lo que los pequeños detalles o diferencias las red StyleGan2 no es capaz de diferenciar.

TABLE II: Matriz de confusión para los cuatro modelos diagnósticos sobre las 2000 imágenes generadas para cada categoría.

	Predicción			
Real	***Benigna***	***H-SIL***	***ASC-US***	***L-SIL***
Benigna	1773	143	78	6
H-SIL	19	1752	127	102
AS-CUS	45	102	1426	427
L-SIL	104	173	585	1138

Si en lugar de analizar la matriz de confusión analizamos los resultados de forma más general, tomando el foco en el porcentaje de precisión en el etiquetado correcto, tenemos que 88,65 de imágenes con células de categoría benigna, el 87.6% de las imágenes de células H-SIL, el 71,3% de las imágenes ASC-US y el 56,9% de las imágenes L-SIL son correctamente etiquetadas. Estos porcentajes pueden apreciarse gráficamente en la Figura 5.

La Figura 6 muestra un conjunto de células creadas sintéticamente y etiquetadas correctamente por el modelo propuesto en este estudio. Como se puede apreciar en dicha figura, las células mostradas no están aisladas, sino que dentro de la imagen de la célula pueden aparecer otro tipo de elementos. Además se aprecian diferentes tonalidades así como diferentes núcleos que representan células dobladas o solapadas con otras. El solapamiento, mezcla y difuminación de células es uno de los factores a considerar por el que el modelo propuesto no ha conseguido una precisión superior de etiquetado. Además, hay que destacar que se han incluido todas las categorías posibles de clasificación de una célula escamosa, ya que en algunos trabajos la categoría ASC-US no es considerada debido a su ambigüedad y esto engañosamente hace aumentar la precisión de los modelos. Sin embargo, tal como se muestra en la Figura 6, con todas las categorías, es como los patólogos expertos encuentran cada una de las células que analizan, de ahí la importancia de trabajar directamente con la imagen real sin realizar ningún tipo de preprocesamiento previo. Esto obtiene la ventaja de poder crear un sistema automático de detección de células basado en la realidad y sin sesgos.

Por último para observar el comportamiento y transición del modelo propuesto para crear una célula, la Figura 4 muestra un total de 6 imágenes del proceso de como una célula comienza a formarse y va evolucionando hasta convertirse en este caso en una célula escamosa de categoría H-SIL.

VI. Conclusiones y trabajos Futuros

El cáncer de cuello uterino es uno de los que más afecta a las mujeres, y su detección precoz es muy importante para superar la enfermedad con el menor número de secuelas posibles. Una de las técnicas de cribado más efectivas para detectar este tipo de enfermedad es la realización periódica de citologías cervicales. El patólogo experto tiene que analizar manualmente las células de la muestra para identificar células patológicas. La idea subyacente de este trabajo es crear un sistema inteligente de detección y clasificación de células, pero existe la problemática de la falta de imágenes reales etiquetadas correctamente para poder crear un modelo computacional efectivo y preciso. Debido a esta problemática, en este trabajo se ha presentado un nuevo enfoque computacional para realizar un etiquetado automático y un aumento sintético de datos para el problema de detección precoz de cáncer de cérvix. El modelo computacional propuesto se basa en la función de una red generativa antagónica junto con una red neuronal convolucional basada en aprendizaje profundo. Los resultados y ajuste de parámetros del modelo han demostrado que el modelo propuesto es capaz de crear un conjunto sintético de datos con una alta precisión para cada una de las cuatro categorías diagnósticas en las cuales se pueden clasificar las células de las muestras. A pesar del buen rendimiento del modelo, una optimización de los hiperparámetros y una evaluación para la mejora del rendimiento del mismo, se presenta como una posibilidad de trabajo futuro. Además, el nuevo modelo mejorado y optimizado permitirá la creación de un conjunto de datos sintético de gran volumen y de mayor calidad, siendo utilizado para entrenar un modelo de clasificación en diferentes categorías diagnosticas. El modelo creado deberá ser validado tanto por expertos patólogos como por diferentes experimentos para demostrar la precisión y robustez para clasificar células de forma automática aumentando la precisión obtenida hasta el momento, es decir, intentando mejorar la precisión humana.

References

[1] Chen, M., Hao, Y., Hwan, K., Wang, L., & Wang, L. (2017). Disease Prediction by Machine Learning Over Big Data From Healthcare Communities. IEEE Access.

[2] Hakama, M., and Rasxnen-Virtanen, U. L. L. A. (1976). Effect of a mass screening program on the risk of cervical cancer. American journal of epidemiology, 103(5), 512-517.

[3] Mohammed M., A., Asmaa, A., and Mohamed Medhat, G. (2020). Classification of COVID-19 in chest X-ray images using DeTraC deep convolutional neural network. Applied Intelligence volume.

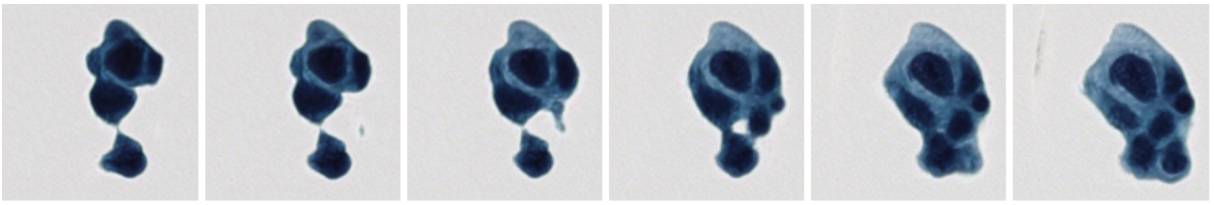


Fig. 4: Creación de imágenes de células generadas por la red StyleGan2 entrenada con el conjunto de células de la categoría H-SIL.

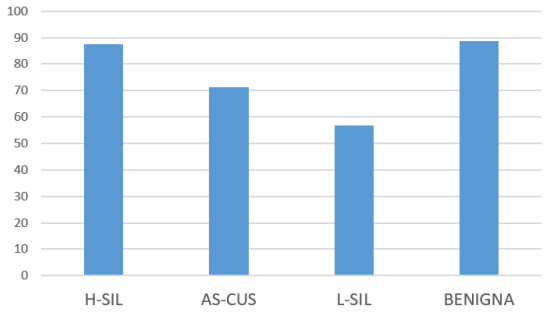
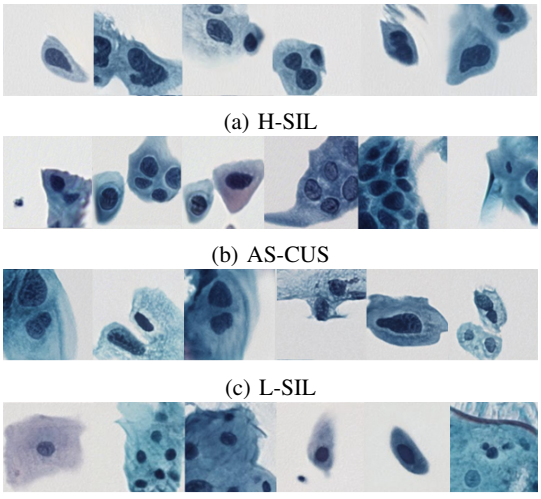


Fig. 5: Imágenes categorizadas correctamente (en %) por el modelo propuesto. Estas imágenes formarán parte de la nueva base de datos sintética.



(a) H-SIL

(b) AS-CUS

(c) L-SIL

(d) Benigna

Fig. 6: Imágenes de células generadas por el modelo propuesto para los diferentes categorías diagnósticas.

[4] Liu, C. (2017). Detecting tuberculosis in chest X-ray images using convolutional neural network. IEEE International Conference on Image Processing (ICIP).

[5] Mahmut, Y., Salman UH, D., and Aykut, E. (2021). mustGAN: multi-stream Generative Adversarial Networks for MR Image Synthesis. Medical Image Analysis.

[6] Papanicolaou, G. N., and Traut, H. F. (1941). The diagnostic value of vaginal smears in carcinoma of the uterus. American Journal of Obstetrics and Gynecology, 42(2), 193-206.

[7] Nayar, R., and Wilbur, D. C. (2015). The pap test and Bethesda 2014. Cancer cytopathology, 123(5), 271-281.

[8] IARC Working Group on the Evaluation of Cancer-Preventive Strategies, International Agency for Research on Cancer, and World Health Organization. (2005). Cervix cancer screening (Vol. 10). Diamond Pocket Books (P) Ltd..

[9] Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., and Aila, T. (2020). Analyzing and improving the image quality of stylegan. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 8110-8119).

[10] Karras, T., Aittala, M., Hellsten, J., Laine, S., Lehtinen, J., and Aila, T. (2020). Training generative adversarial networks with limited data. arXiv preprint arXiv:2006.06676.

[11] Martínez-Más, J., Bueno-Crespo, A., Martínez-España, R., Remezal-Solano, M., Ortiz-González, A., Ortiz-Reina, S., and Martínez-Cendán, J. P. (2020). Classifying Papanicolaou cervical smears through a cell merger approach by deep learning technique. Expert Systems with Applications, 160, 113707.

[12] Sørbye, S. W., Suhrke, P., Revå, B. W., Berland, J., Maurseth, R. J., and Al-Shibli, K. (2017). Accuracy of cervical cytology: comparison of diagnoses of 100 Pap smears read by four pathologists at three hospitals in Norway. BMC clinical pathology, 17(1), 1-6.

[13] Wright Jr, T. C., Stoler, M. H., Behrens, C. M., Sharma, A., Sharma, K., and Apple, R. (2014). Interlaboratory variation in the performance of liquid-based cytology: insights from the ATHENA trial. International journal of cancer, 134(8), 1835-1843.

[14] Kazeminia, S., Baur, C., Kuijper, A., van Ginneken, B., Navab, N., Albarqouni, S., and Mukhopadhyay, A. (2020). GANs for medical image analysis. Artificial Intelligence in Medicine, 101938.

[15] Frid-Adar, M., Diamant, I., Klang, E., Amitai, M., Goldberger, J., and Greenspan, H. (2018). GAN-based synthetic medical image augmentation for increased CNN performance in liver lesion classification. Neurocomputing, 321, 321-331.

[16] Islam, J., and Zhang, Y. (2020). GAN-based synthetic brain PET image generation. Brain informatics, 7, 1-12.

[17] Motlagh N.H., Jannesary M., Aboulkheyr H., Khosravi P., Elemento O., Totonchi M. and Hajirasouliha I. (2018). Breast cancer histopathological image classification: A Deep Learning Approach. bioRxiv, 242818.

[18] Behrmann J., Etmann C., Boskamp T., Casadonte R., Kriegsmann J. and Maab P. (2017). Deep Learning for tumor classification in imaging mass spectrometry. Bioinformatics, 34(7), 1215–1223.

[19] Godinez W.J., Hossain I., Lazic S.E., Davies J.W. y Zhang X. (2017). A multi-scale convolutional neural network for phenotyping high-content cellular images. Bioinformatics, 33(13), 2010-2019.

[20] Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1251-1258).

[21] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition (pp. 248–255).

Selección óptima de variables mediante computación evolutiva para algoritmos de clasificación

Daniel Parra[1]
Universidad Complutense de Madrid
Madrid, España
dparra02@ucm.es

Alberto Gutiérrez[1]
Universidad Complutense de Madrid
Biztools S.L.
Madrid, España
albegu02@ucm.es

Oscar Garnica
Universidad Complutense de Madrid
Madrid, España
ogarnica@ucm.es

Juan Lanchares
Universidad Complutense de Madrid
Madrid, España
julandan@ucm.es

J. Ignacio Hidalgo
Universidad Complutense de Madrid
Madrid, España
hidalgo@dacya.ucm.es

***Abstract*—En este artículo se ha diseñado un sistema de selección de variables (feature selection) para sistemas clasificadores, basado en computación evolutiva. Se han analizado las distintas configuraciones de un algoritmo genético y se propone una estructura particular en el proceso de selección que proporciona resultados interesantes. Se utiliza una codificación binara directa que nos permite realizar la evaluación de los individuos de manera eficiente, en la que se codifica como 1 aquellas variables que serán utilizadas por el clasificador. Para identificar estas variables, se evalúan los individuos mediante el accuracy, obtenido por el clasificador sobre el que se quiere aplicar, en un conjunto de datos reducido. Este artículo se centra en la selección de variables que aporten más información para la correcta clasificación de los usuarios. Para ello se ha utilizado una base de datos sintética de 1000 individuos, que contiene información de los mismos. Nuestro algoritmo evolutivo de selección de variables, se ha aplicado con éxito sobre los algoritmos de Gradient Boosting, llegando hasta valores que rondan el 74% permitiendo incrementar el accuracy en 8 puntos.**

***Index Terms*—Feature Selection, Clasificación, Algoritmo genético, Computación Evolutiva**

I. Introducción

A lo largo de estos últimos años hemos podido presenciar el uso de la inteligencia artificial al servicio de distintos sectores, un posible ejemplo es el empleo de Deep Learning para la diabetes (Zhu et al., 2020). En este estudio se presenta una revisión integral de las diferentes aplicaciones del Deep Learning en el campo de la diabetes, donde podemos encontrar 40 investigaciones. En cada una de ellas se tratan puntos como los modelos empleados, el desarrollo, los resultados y los métodos de referencia para su evaluación. Los principales problemas identificados a lo largo de la investigación fueron la dificultad a la hora de interpretar el modelo y la falta de disponibilidad de datos.

En los problemas de clasificación es común encontrar conjuntos de datos con un gran número de variables, siendo necesario invertir recursos para obtener un subconjunto más manejable (selección de variables), reduciendo la dimensionalidad, los tiempos de ejecución y mejorando los resultados de los modelos. La selección de las variables más adecuadas es uno de los problemas a tratar a la hora de realizar un buen sistema clasificador, ya que se debe conseguir un compromiso entre calidad e interpretabilidad. Utilizar siempre todas las variables no siempre es la mejor forma de afrontar el trabajo, ya que puede introducir sesgos y confusión en los modelos.

En este trabajo se han realizado algoritmos de selección de variables aplicados a un problema de clasificación de individuos, concretamente se propone crear un algoritmo genético que apoye al modelo de clasificación en uso, mediante la selección de las variables con más valor.

A continuación, se describe la organización del artículo:

- Apartado I: Introducción y organización.
- Apartado II: Análisis de trabajos relacionados.
- Apartado III: Descripción y funcionamiento del proyecto.
- Apartado IV: Resultados obtenidos y análisis de los mismos.
- Apartado V: Conclusiones generales y trabajo futuro.

II. Estado del arte

El aprendizaje automático se basa en una premisa muy sencilla, "si introduces basura, recibirás basura" por este motivo es importante elegir sabiamente qué variables tener en cuenta y cuáles descartar. Al trabajar en aplicaciones reales es común encontrar conjuntos de datos que presentan un número reducido de casos con una gran cantidad de

[1]Alberto Gutiérrez y Daniel Parra contribuyeron igualmente a este trabajo como primer autor.

variables, introduciendo así distintas problemáticas como la dimensionalidad y los elevados requisitos de computo (Pirgazi et al., 2019). Para lidiar con estos obstáculos se ha propuesto el uso de técnicas de selección de variables "Feature selection" con las que elegir aquellas que aporten mayor valor. Dentro de la selección de variables podemos encontrar tres ramas en las que categorizar los distintos métodos, filtros, *wrapper* o envolventes y embebidos.

Los métodos de filtrado destacan principalmente por su velocidad y escalabilidad, siendo de gran ayuda en aquellos conjuntos de datos extremadamente grandes. Mediante una serie de procesos estadísticos se asignan puntuaciones a las distintas variables, las mejores serán utilizadas para crear el modelo predictivo. La gran limitación de estos métodos es el hecho de no tener en cuenta la relación entre variables, perdiendo de vista el objetivo de determinar el mejor subconjunto de datos. Dos posibles representantes de esta categoría serían el coeficiente de correlación de Pearson, que nos permite cuantificar la dependencia lineal entre dos variables aleatorias y la información mutua, que busca reducir la incertidumbre de una variable aleatoria X a través del conocimiento de otra variable aleatoria (Guyon and Elisseeff, 2003).

Los métodos wrapper buscan el subconjunto de variables más adecuado para el problema específico, empleando el predictor seleccionado como caja negra a la hora de puntuar los distintos subconjuntos de variables, que son generados a lo largo de las distintas iteraciones. Es considerado el método más costoso desde el punto de vista computacional al necesitar entrenar y probar cada uno de los posibles subconjuntos. Dentro de esta rama encontramos la selección secuencial de variables (sfs), que comienza con un conjunto vacío y va añadiendo de forma individual las distintas variables, buscando aquella que más valor aporte al conjunto. Tras identificar dicha variable, ésta será incorporada de manera permanente al subconjunto, a continuación se pasará a la siguiente iteración, repitiendo el mismo proceso hasta obtener el número de variables deseadas. Partiendo de esta implementación podemos encontrar distintas variantes, como la selección secuencial hacia atrás (sbs), en la que se comenzará con el conjunto de variables completo y se irán eliminado variables o la selección secuencial de avance flotante (sffs), que tiene como base el método sfs e incorpora una componente de vuelta atrás con el sbs (Chandrashekar and Sahin, 2014).

Dentro de los wrapper también podemos encontrar los métodos de búsqueda heurística, en este estudio nos centraremos en los algoritmos genéticos. El objetivo de estos métodos, es la búsqueda de soluciones tomando como inspiración la evolución de los seres vivos. Partiendo de un conjunto de soluciones potenciales se irán introduciendo una serie de cambios en los individuos, a través de cruces o mediante mutaciones. En cada etapa se irá comprobando el desempeño de la población y la condición de finalización, quedándonos con el mejor individuo como solución final.

Por último tenemos los métodos embebidos, estos buscan incorporar la selección de variables como parte del proceso de entrenamiento mientras tratan de combinar las ventajas de los métodos anteriores. Uno de los principales problemas que intenta resolver es alto coste computacional de los métodos wrapper.

Analizando el uso de la informática en la rama de la medicina, podemos encontrar numerosas aportaciones a este sector. Recientemente destaca el uso de redes neuronales para detectar anomalías en imágenes de pacientes, como es el caso del siguiente artículo (Morales-Venegas and Tello-Mijares, 2020), que tratan la identificación del cáncer de mama a partir de imágenes termográficas. En él utilizan un algoritmo de superpíxeles para el preprocesamiento de las imágenes y así obtener una imagen sobre-segmentada, capturando la redundancia de la misma. Tras una serie de procesos con la imagen sobre-segmentada se obtiene una entrada para la red neuronal convolucional. Tras obtener las características mediante la red neuronal se aplicaron 2 algoritmos de clasificación (Random Forest y Voted Perceptron), obteniendo unos resultados de precisión del 100%.

También podemos encontrar ejemplos del uso de algoritmos genéticos enfocados a la selección de variables en el campo de la medicina, los cuales son especialmente interesantes para conjuntos de datos de gran tamaño. En (Kumar et al., 2014), se propone un algoritmo de optimización basado en un algoritmo genético, que permite optimizar los valores de los parámetros de la SVM, obteniendo un subconjunto óptimo de características y mejorando la precisión de la clasificación.

La selección de variables ha resultado ser una herramienta especialmente interesante a la hora de lidiar con conjuntos de datos especialmente grandes. Normalmente, los trabajos que abordan este problema suelen utilizar técnicas de selección de variables incorporadas en bibliotecas de programación del lenguaje correspondiente, es por ello que este trabajo trata de abordar dicha selección mediante la generación propia de un algoritmo genético.

III. Detalles de la implementación

El algoritmo evolutivo tratará de seleccionar las mejores características a la hora de clasificar a los usuarios de un problema concreto. Cada uno de los genes de los cromosomas representarán una variable del problema, el valor 0 indicará que esa característica no se seleccionará y el valor será 1 para el caso contrario. Cada cromosoma estará compuesto de N genes, identificando cada una de las variables del problema original. A su vez cada variable será identificada por la letra G (gen) y su posición en el cromosoma.

A. Población

La generación de la población inicial se hará de manera aleatoria, otra opción sería añadir un individuo con algún gen que conozcamos que funciona especialmente bien como por ejemplo el $G1$, pero en esta ocasión esta opción quedó descartada. Se generarán un total de 50 individuos.

B. Función Fitness

El rendimiento del algoritmo evolutivo depende en gran medida de su *función fitness*, que es única para cada problema

y se encarga de puntuar la aptitud de los individuos. Esta función está estrechamente ligada a la resolución del problema, en nuestro caso consistirá en el entrenamiento de un modelo con las características elegidas y como aptitud, se emplea la tasa de exactitud obtenida al predecir con él. Se entiende por tasa de exactitud a la proporción de aciertos entre todo el conjunto de casos.

C. Selección y reproducción

Para empezar a avanzar entre generaciones, en primer lugar, se calculará la aptitud de cada individuo a través de la función fitness y con este valor se ordenará la población. La primera operación que emplearemos para generar los nuevos individuos será el cruce de un solo punto. Por cada caso se generará un número aleatorio, si este está dentro de la probabilidad de cruce, se procederá a seleccionar los padres para generar un hijo. En este punto se han generado dos variantes en función del método de selección:

- Por un lado, tenemos un **sistema de selección propio (*stud*)** (figura 1), primero, guardaremos en una variable auxiliar un grupo de los 4 mejores candidatos. Los individuos de la población se cruzarán con uno de los candidatos seleccionados con anterioridad a excepción de los dos miembros más aptos, que pasarán limpios a la siguiente generación.

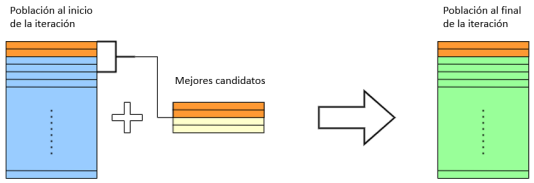


Fig. 1. Relevancia de las variables seleccionadas

- Por el otro lado, se emplea una sencilla implementación del **sistema de selección por torneo**, eligiendo un grupo aleatorio de 5 candidatos y tomando al individuo más apto entre ellos. A continuación, se repite el proceso para seleccionar el sujeto necesario para tener la pareja de padres con el que generar un hijo. Al contrario que el método anterior, este proceso requiere emplear una población auxiliar a la hora de generar los hijos, de lo contrario sobrescribiríamos padres potenciales.

Una vez seleccionados los padres se elige un punto aleatorio y se procede a realizar la mezcla del material genético. Al azar, se elige si la primera parte del hijo será de un padre o del otro.

D. Mutación

Completado el proceso de cruce se realiza una segunda etapa de cambios mediante mutación. De la misma manera que en el caso anterior, se genera un número aleatorio, si está dentro de la probabilidad de mutación se procede a seleccionar un gen y permutar su valor. Este proceso, pese a tener una probabilidad muy baja, es de gran ayuda a la hora de prevenir estancamientos durante la ejecución.

E. Generaciones

El proceso descrito se realiza un número x de veces, en nuestro caso 100 o hasta que los resultados converjan. Obteniendo en la última iteración un resultado cercano a la solución del problema.

IV. Resultados

A. Resultados iniciales

Como en todo problema de aprendizaje automático, inicialmente se realizaron una serie de fases de tratamiento y preporcesado de los datos. Una vez obtenida una fuente de datos limpia, se decidió probar el rendimiento de algunos modelos para este problema mediante validación cruzada (Browne, 2000). Para realizar esta tarea se han empleado los métodos incluidos en la librería para aprendizaje automático Scikit-learn.

Accuracy: 0.70 (+/- 0.11) [Logistic Regression]
Accuracy: 0.68 (+/- 0.08) [BagginC]
Accuracy: 0.69 (+/- 0.10) [GradientBoosting]
Accuracy: 0.69 (+/- 0.10) [RandomForestClassifier]
Accuracy: 0.69 (+/- 0.10) [GaussianNB]
Accuracy: 0.63 (+/- 0.04) [DecisionTreeClassifier]
Accuracy: 0.66 (+/- 0.10) [BernoulliNB]
Accuracy: 0.63 (+/- 0.04) [AdaBoostClassifier]
Accuracy: 0.68 (+/- 0.09) [ExtraTreesClassifier]
Accuracy: 0.70 (+/- 0.11) [Ensemble]

TABLE I
VALIDACIÓN CRUZADA PARA DISTINTOS MODELOS (SIN SELECCIÓN DE VARIABLES)

En nuestro estudio nos centramos en un modelo:

- **Gradient Boosting** (Friedman, 2002): Tasa de exactitud media del 69%
- **Decision Tree** (Myles et al., 2004): Tasa de exactitud media del 63

B. Resultados para Gradient Boosting

*1) **Selección Stud para Gradient Boosting**:* En este caso se empleó el método de invención propia, *Stud*, para la hora de seleccionar a los padres de la siguiente generación. Como presentamos en la siguiente tabla (II), los resultados obtenidos muestran una media de tasa de exactitud 4 puntos superior a los resultados mostrados en la tabla I.

Accuracy Mínimo : 0.71 (+/- 0.10)
Accuracy Máximo : 0.73 (+/- 0.10)
Accuracy Medio : 0.728

TABLE II
APTITUD DE LA POBLACIÓN FINAL (*Gradient Boosting- selección mediante Stud)*

*2) **Selección por torneo para Gradient Boosting**:* En este caso, tabla III, se presentan los resultados obtenidos tras la ejecución del algoritmo evolutivo con una población de 50 individuos y utilizando como función de fitness la exactitud del modelo trabajando con Gradient Boosting.

Accuracy Mínimo : 0.72 (+/- 0.09)
Accuracy Máximo : 0.74 (+/- 0.09)
Accuracy Medio : 0.737

TABLE III
APTITUD DE LA POBLACIÓN FINAL (*Gradient Boosting - selección por torneo)*

Mediante la herramienta SHAP, fue posible generar un gráfico (figura 2) para observar la repercusión de las distintas variables en la salida del modelo.

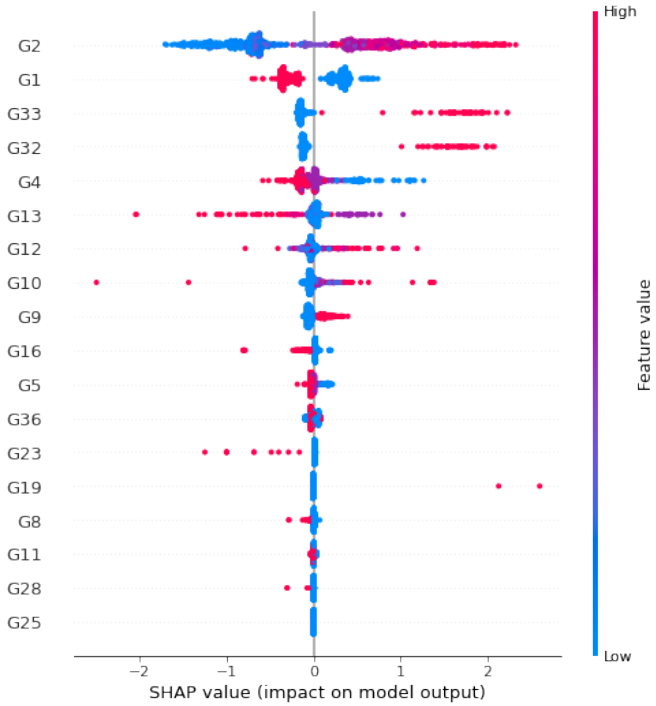


Fig. 2. Relevancia de las variables seleccionadas

Otra funcionalidad que proporciona esta herramienta es la posibilidad de estudiar individuos concretos y analizar en qué se ha basado el modelo para clasificarlos. En primer lugar, figura 3, encontramos un individuo que ha sido clasificado como positivo. En este caso los valores $G2$, $G1$, $G10$, $G9$ son los que nos indican una mayor influencia en la clasificación.

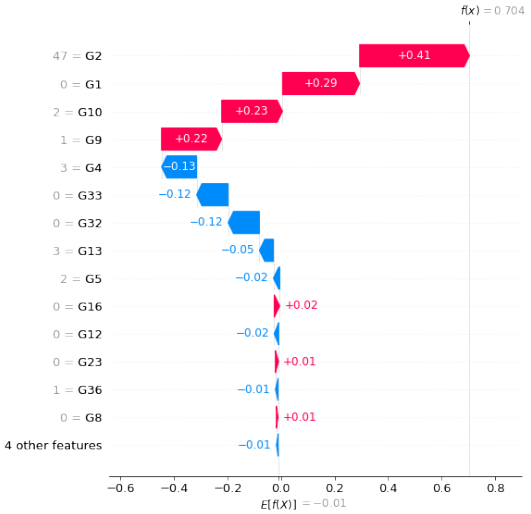


Fig. 3. Relevancia de las variables para un individuo que padece el problema estudiado

Por otro lado se observa en la figura 4 como para otro individuo, en este caso negativo, los factores que determinan su clasificación son: $G2$, $G1$, $G4$, $G33$

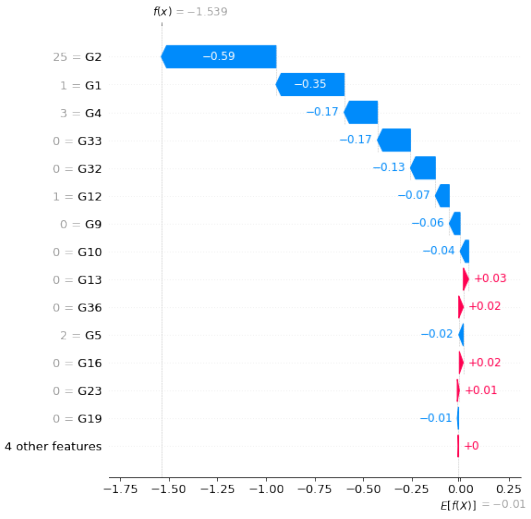


Fig. 4. Relevancia de las variables para un individuo que no padece el problema estudiado

C. Resultados para árboles de decisión

1) ***Selección por torneo para árboles de decisión****:* Para las pruebas con la selección por torneo, de nuevo se utilizó una población de 50 individuos, tabla IV, de esta manera es más difícil que algoritmo converja en las primeras iteraciones. Por otro lado, el candidato con mejor aptitud perdurará a lo largo de las generaciones sin sufrir cambio. Al realizar pruebas con árboles sin emplear las variables del algoritmo evolutivo, los resultados obtenidos estaban por debajo de la media en comparación con el resto clasificadores como se observaba en la tabla I.

Accuracy Mínimo : 0.62 (+/- 0.06)
Accuracy Máximo : 0.69 (+/- 0.06)
Accuracy Medio : 0.686

TABLE IV
APTITUD DE LA POBLACIÓN FINAL (*Árboles de decisión - selección por torneo)*

Accuracy Mínimo : 0.56 (+/- 0.06)
Accuracy Máximo : 0.70 (+/- 0.09)
Accuracy Medio : 0.688

TABLE V
APTITUD DE LA POBLACIÓN FINAL (*Árboles de decisión - selección mediante Stud)*

Teniendo un resultado tan bajo, no era factible considerar válidos los árboles obtenidos con ese modelo. Durante las pruebas del algoritmo evolutivo se optó por tratar de aumentar dicho valor, resultando ser un éxito en corporación con lo anteriormente visto (incremento de 6 puntos en la media).

Llama especialmente la atención la tasa de éxito pese al reducido número de variables que emplea, en comparación con las utilizadas anteriormente.

2) ***Árbol de decisión, método de selección Stud*** *:* De forma complementaria a los resultados del apartado anterior, se decidió repetir el experimento sustituyendo el método de selección por torneo por el desarrollado durante el proyecto. Como se aprecia en la tabla V, los resultados obtenidos superan a los mencionados en la tabla I, incrementado la media de tasa de exactitud en 7 puntos.

Tras comprobar las tasas de exactitud de los diferentes casos se procedió a analizar las variables elegidas por el candidato más apto. Para sorpresa de todos, el mejor resultado se había obtenido con apenas 11 variables.

El árbol obtenido gracias a estas variables (Figura 5) alcanzó una tasa de exactitud del 71%, como contra, tiene un elevado número de falsos negativos, alcanzando un total de 44, frente a los 27 falsos positivos.

```
Correct classification rate: 0.716
              precision    recall  f1-score   support

         0.0       0.72      0.81      0.76       139
         1.0       0.71      0.60      0.65       111

    accuracy                           0.72       250
   macro avg       0.72      0.70      0.71       250
weighted avg       0.72      0.72      0.71       250
```

	tn	fp	fn	tp
0	112	27	44	67

Fig. 5. Resultado mejor candidato, Árbol de decisión con método de selección *Stud*

3) ***Estudio de los resultados****:* A la hora de analizar los resultados obtenidos se realizaron pruebas de convergencia y robustez. Para las primeras se optó por analizar la evolución de los individuos a través de las distintas generaciones. En la tabla VI, se recogen las modificaciones del individuo con mejor desempeño, indicando su *Accuracy* y la ejecución en la que se modificó alguno de sus genes. Para este ejemplo se han utilizados árboles de decisión y el método de selección por torneo.

Accuracy: 0.64 (+/- 0.06) generación: 0
Accuracy: 0.66 (+/- 0.05) generación: 1
Accuracy: 0.66 (+/- 0.08) generación: 3
Accuracy: 0.67 (+/- 0.07) generación: 9
Accuracy: 0.68 (+/- 0.07) generación: 11
Accuracy: 0.68 (+/- 0.06) generación: 12
Accuracy: 0.68 (+/- 0.06) generación: 16
Accuracy: 0.68 (+/- 0.07) generación: 17
Accuracy: 0.69 (+/- 0.06) generación: 24
Accuracy: 0.69 (+/- 0.06) generación: 27
Accuracy: 0.69 (+/- 0.06) generación: 29
Accuracy: 0.69 (+/- 0.06) generación: 34
Accuracy: 0.69 (+/- 0.06) generación: 47

TABLE VI
CAMBIOS DEL MEJOR INDIVIDUO,(*Árboles de decisión - selección por torneo)*

Para comprobar la robustez de los resultados se han realizado un total de 30 ejecuciones, como podemos observar en la siguiente tabla VII los resultados obtenidos son bastante similares, rondando los 68% - 69% de *Accuracy*. En este caso se han utilizados árboles de decisión y el método de selección *Stud*.

Accuracy: 0.68 (+/- 0.10) Ejecución: 0
Accuracy: 0.70 (+/- 0.07) Ejecución: 1
Accuracy: 0.69 (+/- 0.08) Ejecución: 2
Accuracy: 0.69 (+/- 0.09) Ejecución: 3
Accuracy: 0.69 (+/- 0.06) Ejecución: 4
Accuracy: 0.69 (+/- 0.06) Ejecución: 5
Accuracy: 0.69 (+/- 0.10) Ejecución: 6
Accuracy: 0.69 (+/- 0.10) Ejecución: 7
Accuracy: 0.68 (+/- 0.06) Ejecución: 8
Accuracy: 0.68 (+/- 0.06) Ejecución: 9
Accuracy: 0.69 (+/- 0.06) Ejecución: 10
Accuracy: 0.68 (+/- 0.09) Ejecución: 11
Accuracy: 0.68 (+/- 0.07) Ejecución: 12
Accuracy: 0.70 (+/- 0.07) Ejecución: 13
Accuracy: 0.68 (+/- 0.07) Ejecución: 14
Accuracy: 0.67 (+/- 0.08) Ejecución: 15
Accuracy: 0.70 (+/- 0.07) Ejecución: 16
Accuracy: 0.68 (+/- 0.06) Ejecución: 17
Accuracy: 0.68 (+/- 0.08) Ejecución: 18
Accuracy: 0.69 (+/- 0.07) Ejecución: 19
Accuracy: 0.68 (+/- 0.07) Ejecución: 20
Accuracy: 0.68 (+/- 0.06) Ejecución: 21
Accuracy: 0.70 (+/- 0.06) Ejecución: 22
Accuracy: 0.70 (+/- 0.07) Ejecución: 23
Accuracy: 0.67 (+/- 0.07) Ejecución: 24
Accuracy: 0.67 (+/- 0.06) Ejecución: 25
Accuracy: 0.68 (+/- 0.07) Ejecución: 26
Accuracy: 0.66 (+/- 0.06) Ejecución: 27
Accuracy: 0.68 (+/- 0.08) Ejecución: 28
Accuracy: 0.69 (+/- 0.06) Ejecución: 29

TABLE VII
RESULTADOS DE 30 EJECUCIONES, (*Árboles de decisión - selección mediante Stud)*

V. CONCLUSIONES Y TRABAJO A FUTURO

Los resultados del algoritmo evolutivo han sido mejores de lo que se esperaba, pero todavía es necesario seguir mejorando. La exactitud del modelo es importante pero debido a la naturaleza del problema, es más importante la detección de falsos negativos. El modelo debe tener esto presente, por lo que sería interesante desarrollar un algoritmo evolutivo multi-objetivo con el fin de aumentar la exactitud y reducir el número de falsos negativos.

En este trabajo se han desarrollado sistemas de clasificación compuesto por algoritmos evolutivos y clasificadores de los considerados como clásicos en el campo del aprendizaje

automático. Las conclusiones más importantes de este trabajo son:

- Los algoritmos evolutivos diseñados permiten hacer una correcta selección de las variables, proporcionando una mejora en los resultados de los clasificadores, con respecto a su uso clásico, es decir con todas las variables.
- El procedimiento de selección *Stud* que hemos diseñado como una variación sobre los clásicos y que tiene como característica principal el uso de un conjunto elitista que siempre forma parte del proceso de cruce, obtiene mejores resultados que la selección por torneo con presión de selección 5.
- Observando la aptitud de los individuos en cada una de las poblaciones finales, se aprecia cómo todos ellos mantienen una desviación estándar similar, lo que nos indica que los resultados obtenidos son robustos.
- Gracias a los resultados del algoritmo evolutivo, se ha logrado una tasa de mejora de 8 puntos en los árboles de decisión.
- Para el caso de los resultados obtenidos por Gradient boosting tras el uso de las variables elegidas por el algoritmo evolutivo, la mejora obtenida ha sido de 5 puntos.

Las aplicaciones de este trabajo en el futuro son diversas. Se puede utilizar en sistemas de predicción de riesgo de fractura de materiales, o en riesgo de padecer enfermedades como la diabetes o incluso en problemas de predicción de primas en seguros. Todos ellos problemas en los que está involucrado nuestro grupo.

Agradecimientos

Trabajo financiado por Comunidad de Madrid y de los fondos FEDER a través del proyecto de Doctorado Industrial IND2020/TIC-17435, y por el Ministerio de Economía y Competitividad bajo el proyecto RTI2018-095180-B-I00.

References

M. W. Browne. Cross-validation methods. *Journal of Mathematical Psychology*, 44(1):108–132, 2000. ISSN 0022-2496. doi: https://doi.org/10.1006/jmps.1999.1279.

G. Chandrashekar and F. Sahin. A survey on feature selection methods. *Computers & Electrical Engineering*, 40(1):16–28, 2014.

J. H. Friedman. Stochastic gradient boosting. *Computational statistics & data analysis*, 38(4):367–378, 2002.

I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *Journal of machine learning research*, 3 (Mar):1157–1182, 2003.

G. R. Kumar, G. Ramachandra, and K. Nagamani. An efficient feature selection system to integrating svm with genetic algorithm for large medical datasets. *International Journal*, 4(2):272–277, 2014.

D. Morales-Venegas and S. Tello-Mijares. Identificación de cáncer de mama en imágenes termográficas mediante redes neuronales convolucionales. *Revista Ciencia, Ingeniería y Desarrollo Tec Lerdo*, 1(6):–6, 2020.

A. J. Myles, R. N. Feudale, Y. Liu, N. A. Woody, and S. D. Brown. An introduction to decision tree modeling. *Journal of Chemometrics: A Journal of the Chemometrics Society*, 18(6):275–285, 2004.

J. Pirgazi, M. Alimoradi, T. E. Abharian, and M. H. Olyaee. An efficient hybrid filter-wrapper metaheuristic-based gene selection method for high dimensional datasets. *Scientific reports*, 9(1):1–15, 2019.

T. Zhu, K. Li, P. Herrero, and P. Georgiou. Deep learning for diabetes: A systematic review. *IEEE journal of biomedical and health informatics*, PP, 2020.

Calibración probabilística de un modelo de crecimiento del cáncer de vejiga teniendo en cuenta la incertidumbre de los datos

Juan-C. Cortés
Instituto Universitario de Matemática Multidisciplinar
Universitat Politècnica de València
Valencia, España
jccortes@imm.upv.es

Elena López-Navarro
Instituto Universitario de Matemática Multidisciplinar
Universitat Politècnica de València
Valencia, España
ellona1@upvnet.upv.es

Ana Moscardó-García
Instituto Universitario de Matemática Multidisciplinar
Universitat Politècnica de València
Valencia, España
amosgar@alumni.uv.es

Rafael-J. Villanueva
Instituto Universitario de Matemática Multidisciplinar
Universitat Politècnica de València
Valencia, España
rjvillan@imm.upv.es

***Abstract*—El carcinoma vesical o cáncer de vejiga es una de las enfermedades malignas más comunes del sistema urinario y una neoplasia altamente agresiva. Los tumores se pueden extirpar a través de una Resección Transuretral, un procedimiento endoscópico quirúrgico que se considera cirugía menor. Sin embargo, este cáncer se caracteriza por la recurrencia para más de la mitad de los pacientes, esto es, el tumor aparece de nuevo después de un tiempo y puede progresar.**

En este trabajo se propone considerar un modelo lineal de crecimiento tumoral en el que se tenga en cuenta la incertidumbre y los errores en las medidas de los tamaños de los tumores, que se considerarán como variables aleatorias, teniendo en cuenta la reducción de los tamaños tras las resecciones.

Asignando variables aleatorias apropiadas a los tamaños del tumor en diversos instantes temporales, las muestrearemos y calibraremos el modelo usando el algoritmo Particle Swarm Optimization (PSO). Si realizamos este procedimiento un número alto de veces, estimaremos la distribución de probabilidad de los parámetros del modelo al tiempo que las estimaciones de las distribuciones que devuelve el modelo (model output) en los instantes temporales en los que tenemos datos, sean lo más parecidos posible a las variables aleatorias que representan dichos datos.

Una buena calibración permitirá hacer predicciones ajustadas de utilidad para los urólogos en su práctica clínica diaria, conociendo de antemano la evolución del tumor del paciente para administrar tratamientos más adecuados y seleccionar mejores tiempos de seguimiento y revisión.

***Index Terms*—Cáncer de vejiga, Modelo de Crecimiento, Calibración, Cuantificación de la Incertidumbre, Optimización**

I. Introducción

El carcinoma vesical o cáncer de vejiga es una de las enfermedades malignas más comunes del sistema urinario y una neoplasia altamente agresiva [1], [2]. Las estimaciones anuales en todo el mundo son 429.000 nuevos casos y 165.000 muertes. La incidencia es aproximadamente 3 veces mayor en los hombres que en las mujeres. Además, los frecuentes exámenes cistoscópicos resultan ser muy costosos para el sistema de salud pública. De hecho, entre todos los tipos de cáncer, el cáncer de vejiga conlleva el coste más alto por paciente desde el diagnóstico hasta la muerte [3].

En Europa y América del Norte, más del 90% de los cánceres de vejiga son carcinoma urotelial. Estos tumores se estadifican mediante el sistema Tumor-Nodo-Metástasis (sistema TNM): T describe el alcance de la invasión local, N describe si el tumor se ha diseminado a los ganglios linfáticos locales y M describe la diseminación metastásica distante.

Se definen dos grupos principales con resultados clínicos diferentes: el cáncer de vejiga no músculo invasivo (NMIBC, non muscle invasive bladder carcinoma) y el cáncer de vejiga músculo invasivo (MIBC). NMIBC incluye tumores que no han penetrado en la membrana epitelial (estadio Ta), aquellos que invaden la submucosa pero no el músculo (estadio T1), y el carcinoma in situ (CIS).

Alrededor del 80% de los pacientes diagnosticados con este cáncer presentan un carcinoma no invasivo que se puede manejar a través de una Resección Transuretral o RTU, un procedimiento endoscópico quirúrgico para extirpar el tumor de la parte interna de la vejiga [5]. En este procedimiento, se introduce un catéter por la uretra hasta llegar a la vejiga. Con una cámara conectada a la parte superior del catéter, el tumor se encuentra mediante una inspección visual y luego se extrae con un escalpelo. Como no hay necesidad de cortar en el abdomen, se considera una cirugía menor.

Después de la RTU, el tratamiento típico consiste en instilaciones del Bacilo Calmette-Guérin (BCG, bacilo de la tuberculosis) en la vejiga, con el objetivo de que el BCG

Esta investigación ha sido financiada por el proyecto de la Generalitat Valenciana AICO/2020/114.

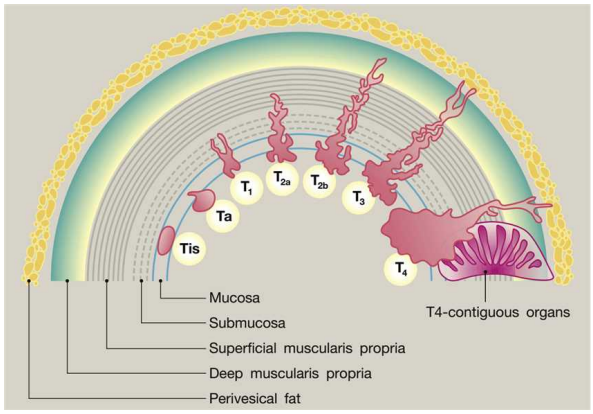


Fig. 1. Tipos de carcinoma vesical según el sistema TNM [4]. Los cánceres no músculo invasivo NMIBC, primeros estadios de la evolución del tumor, corresponden a los tres primeros que aparecen a la izquierda de la figura.

estimule la respuesta inmune del paciente contra el cáncer y entonces, las células cancerosas puedan ser eliminadas [6]. Cuando se administran estas instilaciones, las células de BCG se unen a las células uroteliales y las células cancerosas de la vejiga las internalizan. Debido a esto, las células cancerosas de la vejiga liberan sustancias tales como citoquinas y antígenos que promueven la generación de células inmunitarias contra el cáncer [6].

El cáncer de vejiga se caracteriza por la recurrencia para más de la mitad de los pacientes: el tumor aparece de nuevo después de un tiempo y puede progresar para convertirse en un cáncer músculo-invasivo. Es entonces cuando los tratamientos se vuelven más agresivos, incluyendo la extirpación de la vejiga para evitar la propagación del cáncer a otras partes del cuerpo [7].

Sin embargo, a pesar de los avances en las técnicas de biología molecular [8] y el conocimiento de las etapas del cáncer y los tratamientos [9], [10], todavía no se conocen los mecanismos de la evolución del cáncer de vejiga en pacientes particulares, por qué y cuándo algunos de ellos se curan y otros tienen recurrencias. De hecho, los protocolos de tratamiento esencialmente no han cambiado en los últimos 25 años.

Con la idea de mejorar el conocimiento de la evolución del cáncer, en los últimos tiempos se vienen desarrollando modelos matemático-computacionales que permiten simular dicha evolución [11]. En esta línea, nuestro objetivo es considerar un modelo sencillo de crecimiento tumoral que pueda ayudar a los urólogos en su práctica clínica diaria, que les permitan tener un conocimiento lo más ajustado y personalizado de la evolución de la enfermedad en cada paciente.

Para esto consideraremos los datos proporcionados de la evolución y tratamientos aplicados a un paciente determinado. Habitualmente, los tamaños de los tumores se estiman mediante evaluación visual, por lo que los datos proporcionados estarán sometidos a errores (incertidumbre).

El modelo que vamos a usar será un modelo sencillo de crecimiento lineal de primer orden cuyos parámetros habrá que calibrar teniendo en cuenta la incertidumbre de los datos y las RTUs realizadas al paciente. Así, el modelo describirá la dinámica del cáncer de vejiga del paciente, de modo que si conseguimos una buena calibración, podríamos predecir la evolución de su enfermedad. Este enfoque novedoso en el mundo del cáncer de vejiga, abre la posibilidad de conocer la evolución de la enfermedad de un paciente particular, determinar con tiempo fechas de controles y seguimientos, planificar las RTUs o cuál es el mejor momento para aplicar un cierto tratamiento, como las instilaciones de BCG.

II. Datos

En la Comunidad Valenciana, un paciente acude al médico de familia o a las urgencias del hospital cuando empieza a orinar sangre (hematuria). Entonces, el paciente se dirige al urólogo, quien, mediante ecografías o una citoscopia, diagnostica si hay un tumor dentro de la vejiga. Si es así, el paciente es tratado por los médicos del Departamento de Urología. En nuestro caso, hemos trabajado con médicos pertenecientes al Hospital Universitari i Politècnic La Fe. Los médicos recogen tejidos biológicos de pacientes que padecen cáncer de vejiga de un estadio no músculo-invasivo. Cuando, por el tamaño del tumor, se estima que hay que intervenir y resecarlo, el protocolo habitual es

- realizar una RTU al paciente;
- tras la RTU, el médico especialista en anatomía-patología, observa el tumor, mide el tamaño del mismo (diámetro) y la respuesta inmune en el microambiente tumoral mediante el recuento de células inflamatorias (CD3, CD20, CD56, CD68, CD138, FoxP3 y triptasa). Con los datos recogidos, el médico anatomopatólogo informa al urólogo para que esté al corriente de la situación del paciente;
- administrar instilaciones de BCG;
- realizar citologías y cistoscopias para control de seguimiento y revisiones.

En caso de recidiva (recurrencia del cáncer de vejiga), los médicos repiten el ciclo descrito arriba hasta que el paciente esté completamente curado o el cáncer haya aumentado su tamaño e invadido la capa muscular de la vejiga. En este último caso, hay que aplicar tratamientos más agresivos, como hemos mencionado anteriormente.

De entre los pacientes disponibles, hemos elegido uno de la base de datos del Departamento de Urología del Hospital Universitari i Politècnic La Fe para nuestro propósito. Los datos disponibles de este paciente se recogen y resumen en la Tabla I tal cual aparecen en las notas facilitadas.

Como ya hemos indicado, la medición realizada del diámetro del tumor suele ser visual. Aunque los errores en la apreciación visual no suelen ser determinantes para que el urólogo no tenga una idea bastante clara sobre la evolución del cáncer, a efectos del estudio que vamos a realizar, asumiremos: 1) si el tamaño viene dado por un rango, tomaremos la media como tamaño del tumor; 2) un error de medida del 25% del valor observado, esto es, $\sigma_t^2 = 0.25\mu_t$, donde μ_t es el valor medido del diámetro del tumor (Tabla II), para $t = t_0, \ldots, t_5$.

Para cuantificar la incertidumbre de los datos, vamos a considerar que los datos son una variable aleatoria normal

TABLE I
DATOS CORRESPONDIENTES DE UN PACIENTE QUE FUE DIAGNOSTICADO A PRIMEROS DE MARZO DE 2012. DESDE ENTONCES, SE LE HAN PRACTICADO TRES RTU Y HA SIDO TRATADO CON TRES SESIONES DE INSTILACIONES DE BCG, CADA UNA DE ELLAS DESPUÉS DE CADA RTU.

Instante temporal	**Día**	**Fecha**	**Procedimiento**	**Diámetro μ_t del tumor**
$t = t_0$	$t = 0$	01/Mar/2012	Ecografía	3-5 mm
$t = t_1$	$t = 105$	14/Jun/2012	RTU	25 mm
$t = t_2$	$t = 1081$	15/Feb/2015	Citoscopia	1-2 mm
$t = t_3$	$t = 1153$	28/Abr/2015	RTU	5 mm
$t = t_4$	$t = 1796$	30/Ene/2017	Citoscopia	20 mm
$t = t_5$	$t = 1839$	14/Mar/2017	RTU	30-35 mm

(Gaussiana) de media el valor del dato y la varianza el error del 25%. En la Tabla II hay un resumen de los datos, el error asumido y la variable aleatoria que modeliza la incertidumbre de los datos.

TABLE II
MEDIA, VARIANZA Y VARIABLE ALEATORIA QUE MODELA LA INCERTIDUMBRE DE LOS DATOS DE LA TABLA I.

t	Media μ_t	Varianza σ_t^2	Distribución normal
$t = 0$	4	1	$N(4, 1)$
$t = 105$	25	6.25	$N(25, 6.25)$
$t = 1081$	1.5	0.375	$N(1.5, 0.375)$
$t = 1153$	5	1.25	$N(5, 1.25)$
$t = 1796$	20	5	$N(20, 5)$
$t = 1839$	32.5	8.125	$N(32.5, 8.125)$

III. MODELO

Vamos a usar un modelo lineal discreto de primer orden para modelizar el crecimiento del tumor. Como hemos visto, el crecimiento del tumor se interrumpe cada vez que hay una RTU, cuando empieza a crecer nuevamente partiendo de una nueva situación inicial. Por esto, aunque el modelo de crecimiento sea el mismo, tras cada RTU la condición inicial cambia, lo que nos obliga a definir el modelo en 3 partes.

Tomaremos el paso temporal t de un día, y sea $T(t)$ el diámetro del tumor el día t, y como aparece en la Tabla I, $t_0 = 1$ de marzo de 2012, $t_1 = 14$ de junio de 2012, $t_3 = 28$ de abril de 2015 y $t_5 = 14$ de marzo de 2017. Llamaremos t_1^-, t_3^-, t_5^- a los instantes justo antes de la RTU y t_1^+, t_3^+, t_5^+ justo después de la RTU. Así, el crecimiento del tumor seguirá las ecuaciones

$$T(t+1) = T(t) + k_1 T(t),\ T(t_0) = T_0, \qquad t \in [t_0, t_1^-], \quad (1)$$
$$T(t+1) = T(t) + k_2 T(t),\ T(t_1^+) = T_1, \qquad t \in [t_1^+, t_3^-], \quad (2)$$
$$T(t+1) = T(t) + k_3 T(t),\ T(t_3^+) = T_2, \qquad t \in [t_3^+, t_5^-], \quad (3)$$

donde k_i y T_i, $i = 1, 2, 3$, son las tasas de crecimiento en cada período y los diámetros iniciales de los tumores para cada período, respectivamente.

En el modelo (1)-(3), T_0 es la variable aleatoria $N(4, 1)$. Sin embargo, k_1, k_2, k_3 y T_1, T_2 son variables aleatorias desconocidas que tendremos que estimar usando una técnica computacional que precisará de repetidos calibrados mediante el algoritmo Particle Swarm Optimization (PSO).

IV. CALIBRADO PROBABILÍSTICO

Para estimar las variables aleatorias desconocidas, lo haremos por partes.

1) Primer período entre RTUs, $[t_0, t_1^-]$, con la ecuación (1), donde estimaremos k_1 conociendo T_0 y $T(t_1^-)$.
2) Segundo período entre RTUs, $[t_1^+, t_3^-]$, con la ecuación (2), habrá que determinar k_2 y T_1 sabiendo $T(t_2)$ y $T(t_3^-)$.
3) Tercer período entre RTUs, $[t_3^+, t_5^-]$, con la ecuación (3), de forma análoga, tendremos que estimar k_3 y T_2 sabiendo $T(t_4)$ y $T(t_5^-)$.

El método de estimación en la primera parte, que es el más sencillo será como sigue. La solución de la ecuación discreta $T(t+1) = T(t) + k_1 T(t),\ T(t_0) = T_0$ es $T(t) = T_0 k_1^t$. Si en $t = 105$ el valor de $T(105) = T_{105}$, podemos despejar k_1 como

$$k_1 = \sqrt[105]{\frac{T_{105}}{T_0}} \quad (4)$$

Así, si obtenemos K pares de muestras (m_0, m_1) de las variables aleatorias conocidas T_0 y $T(t_1^-)$, usando la expresión (4) tendremos K valores del parámetro $k_1 = \sqrt[105]{\frac{m_1}{m_0}}$, de los cuales podremos obtener una estimación de la variable aleatoria k_1.

Aunque el razonamiento aplicado para la primera parte podría ser extrapolable a la segunda y tercera parte, vamos a usar una técnica diferente donde precisaremos del algoritmo PSO para realizar un calibrado probabilístico. Este cambio de técnica lo hacemos porque queremos comprobar cómo funciona esta técnica en un modelo sencillo para el caso en que cuando se desee aplicar a modelos más complejos donde la solución sea desconocida, tengamos ciertas garantías de que pueda devolver resultado ajustados. Debemos mencionar que, aunque el modelo es sencillo, el hecho de que sea un modelo exponencial puede dificultar el calibrado por la sensibilidad de los parámetros, esto es, pequeños cambios en los parámetros del modelo puede convertirse en cambios exponenciales en el output del modelo.

La estimación de los parámetros de la segunda parte se realiza como sigue. Aquí, obtendremos K pares de muestras (m_2, m_3) de las variables aleatorias conocidas T_2 y $T(t_3^-)$. Calibraremos el modelo, utilizando PSO, para obtener K valores del parámetro k_2 y de la condición inicial en el instante t_1^+ de forma que

$$F(k_2) = |T(t_2) - m_2| + |T(t_3^-) - m_3| \quad (5)$$

sea lo mínimo posible. Debemos tener en cuenta que la fórmula del error (5) aproxima mejor los valores más grandes. Esto está pensado a propósito, porque deseamos mejor precisión cuando el tumor es grande, ya que es cuando la decisión de realizar la RTU debe tomarse.

Para la tercera parte, la técnica es análoga a la segunda, pero las muestras (m_4, m_5) se obtienen de T_4 y $T(t_5^-)$, para

obtener K valores del parámetro k_3 y de la condición inicial en el instante t_3^+ que minimice $|T(t_4) - m_4| + |T(t_5^-) - m_5|$.

El objetivo es obtener una gran cantidad de muestras de datos para cubrir al máximo todas las combinaciones probables de crecimiento del tumor en cada período entre RTUs. Luego, con la calibración, se espera capturar la incertidumbre de los datos dada por las muestras e, indirectamente, obtener los valores de los parámetros del modelo que estiman su distribución de probabilidad como variables aleatorias. Este último aspecto es muy novedoso, porque habitualmente, los autores asumen conocida la variable aleatoria que representa la incertidumbre de los parámetros.

El algoritmo de PSO utilizado es una versión estándar con una aleatoriedad del 5%, esto es, que el 5% de las partículas actualizadas se descartarán y se sustituirán por una partícula aleatoria nueva. Además, controlamos que las partículas no se salgan del espacio de parámetros determinado.

V. Resultados

En los tres períodos, hemos muestreado $K = 1000$ veces las distribuciones de probabilidad normales de los datos en $t = 0$ y $t = 105$ (Tabla II). En el primer período, hemos obtenido los valores de k_1 mediante la expresión (4) y su histograma puede verse en la gráfica de la izquierda de la Fig. 2.

Para el segundo período hemos calibrado con PSO $K = 1000$ muestras de las distribuciones de probabilidad de los datos para $t = t_2 = 1081$ y $t = t_3 = 1153$. En cada PSO realizamos 1000 generaciones. El histograma de la tasa de crecimiento k_2 aparece en la gráfica de en medio de la Fig. 2.

Y en el tercer período, de forma análoga para $t = t_4 = 1796$ y $t = t_5 = 1839$. El histograma de la tasa de crecimiento k_3 aparece en la gráfica de la derecha de la Fig. 2. La media de k_1 es 1.0175, la de k_2 es 1.0172 y la de k_3 es 1.0105, con lo que puede apreciarse un descenso en las medias de las tasas de crecimiento cada vez que se le hace al paciente una RTU. Sin embargo, el histograma para k_3 es binodal, lo que parece indicar que el tamaño del tumor puede tener una tendencia decreciente y quizás desaparecer, o de tendencia creciente y pasar a un estadio más agresivo. Siguiendo con esta idea y como las distribuciones de k_1, k_2 y k_3 tienen solapes, esto puede reflejar la incertidumbre existente sobre la evolución del tamaño del los tumores y el hecho mencionado en la introducción sobre que la mitad de los pacientes tienen tumores recurrentes.

En los períodos 2 y 3, también ajustamos las correspondientes condiciones iniciales $t_1^+ = 105$ y $t_3^+ = 1153$ tras la RTU. Sus histogramas de densidad pueden verse en la Fig. 3. Aunque la probabilidad se acumula en los extremos, el de la izquierda es cercano a cero, lo que indica que la probabilidad de que el urólogo haya hecho un buen trabajo eliminando todo lo posible el tumor, es alta. La acumulación en los extremos de la izquierda puede explicarse por la sensibilidad que puede tener el modelo respecto a la condición inicial, porque la variación es de 0.1 milímetros en el tamaño inicial.

En las Figs. 4 y 5 se puede apreciar el ajuste conseguido con la aplicación de la técnica descrita. En la Fig. 4 con las funciones de densidad de probabilidad de los datos y el histograma del output del modelo, y en la Fig. 5 , con las funciones e histograma acumulados.

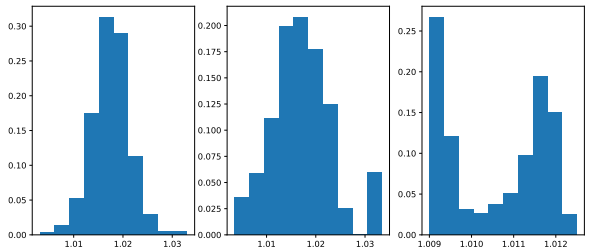


Fig. 2. Histogramas normalizados de las tasas de crecimiento construidas con los $K = 1000$ calibrados del modelo. k_1 tasa del primer período, gráfica de la izquierda; k_2 tasa del segundo período, gráfica de en medio; k_3 tasa del tercer período, gráfica de la derecha. Las medias de los k_1, k_2 y k_3 son 1.0175, 1.0172 y 1.0105, respectivamente, que refleja un descenso en la tasa de crecimiento tras cada RTU.

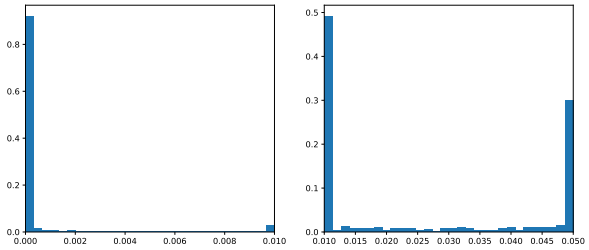


Fig. 3. Histogramas normalizados de lo que queda de tumor vesical tras las RTUs para $t_1^+ = 105$ y $t_3^+ = 1153$.

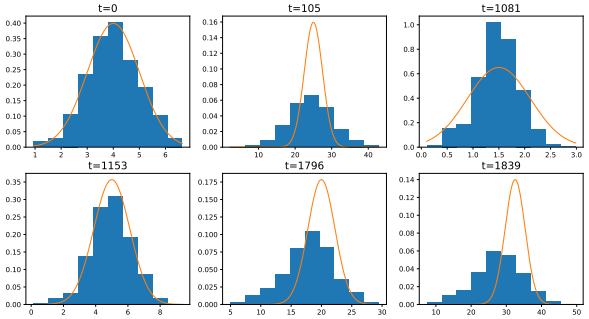


Fig. 4. Comparación entre los histogramas generados por las salidas del modelo y la función de densidad de probabilidad asignada a los datos (Tabla II).

Si bien la aproximación es razonable, puede apreciarse que tomar $K = 1000$ no parece ser suficiente para poder tener una buena aproximación y deberíamos haber tomado más. No obstante, la técnica propuesta parece prometedora, y esperamos mejoras sustanciales en las aproximaciones aumentando el valor de K.

VI. Conclusión

En este trabajo planteamos un modelo de crecimiento de cáncer vesical donde consideramos la reducción, y posible desaparición, del tumor mediante la intervención conocida

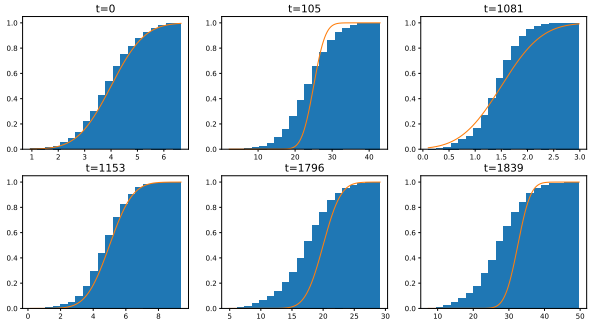


Fig. 5. Comparación entre los histogramas acumulados generados por las salidas del modelo y la función de densidad de probabilidad acumulada asignada a los datos (Tabla II).

como Resección Transuretral o RTU. Las RTUs generan saltos en la continuidad del crecimiento del cáncer, por lo que hemos de considerar el modelo dividido en los 3 períodos entre RTUs.

A partir de aquí, con los datos disponibles y su error intrínseco, hemos asignado variables aleatorias a esos datos usando distribuciones normales (Gaussianas). Así, hemos diseñado una técnica probabilística para el calibrado del modelo utilizando una versión del algoritmo Particle Swarm Optimization (PSO). Esta técnica consiste en muestrear de las distribuciones de los datos, verosímiles tamaños del tumor determinados por las distribuciones normales y calibrar el modelo. Si repetimos este proceso de muestreo-calibración un número suficiente de veces, nos aseguramos que el modelo se acerque en los instantes en los que tenemos datos a sus distribuciones de probabilidad, al tiempo que obtenemos un estimación de la distribución de probabilidad de los parámetros, cuestión de máximo interés, pero poco considerada en la literatura.

Los cálculos realizados para el calibrado nos podrían permitir hacer predicciones sobre la evolución del tumor tras la tercera RTU y estimar cuándo se alcanzarán diámetros de tumor que deban ser preocupantes o, al menos, deban ser revisados.

Aunque el resultado obtenido parece prometedor, el número de iteraciones de PSO cómo el número de muestras que hay que calibrar, parecen factores importantes en los que habrá que alcanzar un equilibrio que no aumente mucho el coste computacional y tener resultados de calidad. Por otra parte, esta técnica debe probarse con modelos más complejos donde el calibrado y el tratamiento no sean tan sencillos para ver como funciona el algoritmo y valorar si es necesario realizar sobre los valores obtenidos de los parámetros del modelo, alguna selección.

References

[1] "Official Site for Spanish Medic Oncology Society," https://www.seom.org, accessed: 25/09/2018.

[2] R. T. Greenlee, M. B. Hill-Harmon, T. Murray, and M. Thun, "Cancer statistics, 2001," *CA: A Cancer Journal for Clinicians*, vol. 51, no. 1, pp. 15–36, Jan 2001. [Online]. Available: https://doi.org/10.3322/canjclin.51.1.15

[3] E. B. Avritscher, C. D. Cooksley, H. B. Grossman, A. L. Sabichi, L. Hamblin, C. P. Dinney, and L. S. Elting, "Clinical model of lifetime cost of treating bladder cancer and associated complications," *Urology*, vol. 68, no. 3, pp. 549–553, Sep. 2006. [Online]. Available: https://doi.org/10.1016/j.urology.2006.03.062

[4] M. Yazbek-Hanna, P. Whelan, and S. Jain, "Bladder cancer," *Medicine*, vol. 44, no. 1, pp. 52–55, Jan. 2016. [Online]. Available: https://doi.org/10.1016/j.mpmed.2015.10.008

[5] R. Devere White and E. Stapp, "Predicting prognosis in patients with superficial bladder cancer," *ONCOLOGY*, vol. 12, no. 12, pp. 1717–1723, 1998. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/9874845/

[6] G. Redelman-Sidi, M. S. Glickman, and B. H. Bochner, "The mechanism of action of BCG therapy for bladder cancer—a current perspective," *Nature Reviews Urology*, vol. 11, no. 3, pp. 153–162, feb 2014. [Online]. Available: https://doi.org/10.1038/nrurol.2014.15

[7] S. Holmang, H. Hedelin, C. Anderstrom, and S. L. Johansson, "The relationship among multiple recurrences, progression and prognosis of patients with stages TA and t1 transitional cell cancer of the bladder followed for at least 20 years," *The Journal of Urology*, vol. 153, no. 6, pp. 1823–1827, jun 1995. [Online]. Available: https://doi.org/10.1016/s0022-5347(01)67321-x

[8] F. Audenet, K. Attalla, and J. P. Sfakianos, "The evolution of bladder cancer genomics: What have we learned and how can we use it?" *Urologic Oncology: Seminars and Original Investigations*, mar 2018. [Online]. Available: https://doi.org/10.1016/j.urolonc.2018.02.017

[9] "Bladder Cancer Treatment (PDQ)-Health Professional Version," https://www.cancer.gov/types/bladder/hp/bladder-treatment-pdq, accessed: 25/09/2018.

[10] "Bladder Cancer Treatment (PDQ)-Patient Version," https://www.cancer.gov/types/bladder/patient/bladder-treatment-pdq, accessed: 25/09/2018.

[11] H. Enderling and K. A. Rejniak, "Simulating cancer: Computational models in oncology," *Frontiers in Oncology*, vol. 3, 2013. [Online]. Available: https://doi.org/10.3389/fonc.2013.00233

Heurística para el desarrollo de un sistema experto epidemiológico aplicado al COVID-19

Beatriz González-Pérez
Universidad Complutense de Madrid
Instituto de Matemática Interdisciplinar
beatrizg@ucm.es

Concepción Nuñez
Instituto de Investigación Sanitaria.
Hospital Clínico San Carlos
conchita.npardo@gmail.com

José L. Sánchez
Universidad Complutense de Madrid
Madrid, España
joseluis.sanchezmaronas@gmail.com

Gabriel Valverde
Universidad Complutense de Madrid
Madrid, España
gvalverd@ucm.es

José Manuel Velasco
Universidad Complutense de Madrid
Madrid, España
mvelascc@ucm.es

***Abstract*—Desarrollamos dos modelos para el seguimiento y la previsión en tiempo real de la evolución de la pandemia COVID-19: un modelo de regresión no lineal y un modelo de corrección del error. Estos modelos se optimizan utilizando diferentes heurísticas y conjuntamente constituyen un sistema experto epidemiológico. este sistema permite detectar los picos de la pandemia, hacer previsiones a corto y largo plazo del número de infectados, de las muertes, de las personas que requieren hospitalización y de las personas que ingresan en cuidados intensivos. El sistema está dotado de un procedimiento de control para detectar los cambios de tendencia y definir el final de una ola y el comienzo de otra. Además, el modelo de regresión no lineal depende sólo de cuatro parámetros por serie, lo que facilita la interpretación y el seguimiento a lo largo del tiempo de cada variable por parte de los responsables médicos. Esta característica tmabién permite estudiar el efecto de las intervenciones de las autoridades sanitarias a lo largo del tiempo para aconsejar cómo proceder en futuros brotes. El modelo de corrección del error desarrollado trabaja con cointegración entre series y tiene una gran capacidad de previsión. Nuestro sistema está preparado para trabajar en paralelo en todas las comunidades autónomas de España. En este trabajo haremos una comparativa entre nuestros modelos y una extensión del modelo SIR con un estado que representa el confinamiento (SCIR).**

***Index Terms*—Artificial Intelligence; Machine Learning; Nonlinear Regression; Error Correction Model; SIR**

I. Introducción

La enfermedad por coronavirus 2019 (COVID-19), causada por el llamado virus SARS-CoV-2, se ha extendido por todo el mundo provocando una terrible pandemia. Comenzando en China en diciembre de 2019, los siguientes dos países fueron Italia y España. La alta transmisibilidad de la infección hizo que algunas regiones sufrieran un especial impacto. Tal es el caso de la Comunidad Autónoma de Madrid en España. El 11 de marzo, la Organización Mundial de la Salud (OMS) declaró la COVID-19 como pandemia. En esa misma fecha, Madrid se encontraba ya en una situación de extrema gravedad y se cerraron todos los centros educativos. Tres días después, el 14 de marzo, se declaró el estado de alarma en todo el país. A parit de ahí, las autoridades han tomado medidas de diversa magnitud a lo largo de las diferentes oleadas que se han sucedido hasta ahora. Los modelos matemáticos para seguir los cambios en el comportamiento y el patrón de infección parecen ser herramientas esenciales para tomar decisiones futuras.

Algunos trabajos como [1] o [2] fueron pioneros en recoger soluciones basadas en la inteligencia artificial y el conocimiento experto, entre los miles de artículos publicados sobre el tema.

El modelo SIR (”Susceptible”, ”Infectado”, ”Recuperado”) y sus extensiones son modelos epidemiológicos tradicionales. Se trata de modelos compartimentados de ecuaciones diferenciales que relacionan las variaciones de los distintos grupos de población (compartimentos) a través de la tasa de infección y el periodo infeccioso medio. La mayoría de los estudios recientes se basan en modificaciones del modelo SIR (véase [3], [4] y [5], entre otros). La idea subyacente es modelar las ondas de una pandemia como aumentos y disminuciones exponenciales a la izquierda y a la derecha de un pico de máxima incidencia. A efectos comparativos implementamos la extensión del modelo SIR desarrollada por [6], SCIR: un modelo SIR con confinamiento.

Nosotros abordamos el problema desde otro punto de vista: los modelos predictivos de regresión no lineal. Investigadores de la Escuela Andaluza de Salud Pública de Granada han desarrollado un modelo predictivo de la epidemia de COVID-19 en España con un ajuste a una curva de Gompertz [7]. Para el ajuste de la curva de Gompertz a los datos acumulados observados de casos y muertes, utilizaron los algoritmos de Nelter-Mead [8] implementados por Nash [9]. El software utilizado para los cálculos fue R a través del paquete drc.

Nuestra estrategia amplía este enfoque al permitir una mayor flexibilidad en el ajuste de las curvas de Gompertz, especialmente en las colas de la distribución. Otra aproximación de Gompertz es [10]. Nuestro sistema experto elige automáticamente el mejor ajuste entre una variedad de modelos, incluyendo también las curvas de Gauss, doble exponencial y doble Pareto. Además, toda la programación, el algoritmo de optimización y la heurística son originales.

Además, desarrollamos un modelo de corrección de errores

(MCE). Esta aproximación pertenece a una categoría de modelos de series temporales múltiples para datos en los que las variables subyacentes tienen una tendencia estocástica común a largo plazo, también conocida como cointegración.

Nuestro grupo de investigación se inscribió en la "Acción matemática contra los coronavirus", se ha construido una iniciativa de predicción cooperativa -un meta-predictor- para proporcionar a las autoridades información sobre el comportamiento a corto plazo de variables de gran interés en la propagación del virus COVID-19. El método utiliza las predicciones de diferentes modelos/algoritmos, aportados por los investigadores participantes, y construye combinaciones optimizadas de los mismos, desagregadas por Comunidades Autónomas del Comité Español de Matemáticas (CEMAT). Dentro de esta iniciativa hemos participado junto con otros grupos de investigación en la acción "Predicción Cooperativa" [11], proporcionando predicciones diarias con nuestro modelo preliminar desde marzo de 2020, durante toda la primera ola de la pandemia . Es precisamente gracias a este trabajo colaborativo que nuestro modelo ha sido validado.

Los resultados obtenidos en este trabajo son reproducibles utilizando el código de nuestro repositorio público: https://github.com/mikiNadal/covid19_article_reproducible.

La sección II presenta el modelo de regresión no lineal. La sección III presenta el modelo de corrección de errores. La sección IV presenta el modelo SCIR. La sección V compara los tres modelos con diferentes métricas. Por último, la sección VI incluye las conclusiones.

II. Modelo de regresión No Lineal

Nuestro objetivo es desarrollar un marco teórico que permita detectar picos y hacer un seguimiento y previsión a corto y largo plazo del número de infectados, personas que requieren hospitalización y muertes durante una enfermedad infecciosa. Este modelo se implementa con un sistema experto de Inteligencia Artificial basado en regresión no lineal y de gran utilidad para conocer la eficacia de las intervenciones impulsadas por los gobiernos y aconsejar cómo proceder en futuros brotes. Además, el algoritmo de Machine Learning desarrollado permite la ejecución en paralelo y la introducción de nuevos datos en tiempo real, y es escalable.

Nuestro modelo se basa en la estimación directa de la función de distribución de cada una de las series en estudio y en la dualidad entre la función de distribución y la función de densidad. Dado que estas dos funciones caracterizan completamente la distribución de probabilidad de una variable continua, nuestro modelo es capaz de capturar las principales características de los brotes epidémicos. A ello se añade su simplicidad, ya que se formula sólo mediante tres parámetros. En adelante, nos referiremos a nuestra primera propuesta de modelo epidemiológico como modelo MATGEN en honor a nuestro grupo inscrito en el *Acción matemática contra los coronavirus* [11], una iniciativa del CEMAT (Comité Español de Matemáticas).

La notación empleada en este trabajo es la siguiente:

Sea la conocida función de densidad de una variable normal de media μ y varianza σ^2, $N(\mu, \sigma)$, dada por

$$f\left(t\right) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{t-\mu}{\sigma}\right)^2}$$

and both its distribution function and the right tail as follows:

$$F\left(t\right) = \int_{-\infty}^{t} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} dx$$

$$1 - F\left(t\right) = \int_{t}^{\infty} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} dx$$

Las series de datos de COVID-19 en España incluyen día a día el número acumulado de personas infectadas, personas que requieren hospitalización y muertes. Estos datos pueden descargarse de [12].

Denotamos las frecuencias relativas y acumulativas en el tiempo t como sigue

$$N_t \text{ acumulada por día,}$$

$$n_t = N_t - N_{t-1} \text{ nuevos casos por día,}$$

$$f_t = \frac{n_t}{n},$$

$$F_t = suma_{i=1}^{t} f_i,$$

donde n es el número total de casos al final de la pandemia.

Además, introducimos la media de las frecuencias acumuladas en el tiempo t dada por

$$Av_t = \frac{1}{t}\sum_{i=1}^{t} F_i.$$

En este contexto, vamos a trabajar con el siguiente modelo de regresión no lineal

$$F_i = F\left(i\right) + \varepsilon_i, \varepsilon_i \sim N\left(0,\ \tau\right),\ i = 1, \cdots, t,$$

donde los parámetros n, μ y σ se van a estimar por el método de mínimos cuadrados.

Para una introducción a la regresión frecuentista y bayesiana, véase [13].

A. Algoritmo para la detección del pico

Inicializamos t en t_0, el momento actual.

Calculamos el error medio cuadrático de la forma siguiente:

$$ECM\left(t, n, \mu, \sigma | n_1, \cdots, n_t\right) = \frac{1}{t}\sum_{i=1}^{t}\left(F(i) - F_i\right)^2,$$

la varianza total

$$SCT\left(t, n, \mu, \sigma | n_1, \cdots, n_t\right) = \frac{1}{t}\sum_{i=1}^{t}\left(F_i - Av_t\right)^2,$$

y el coeficiente de determinación

$$R^2\left(t,n,\mu,\sigma|n_1,\cdots,n_t\right)=1-\frac{ECM\left(t,n,\mu,\sigma|n_1,\cdots,n_t\right)}{SCT\left(t,n,\mu,\sigma|n_1,\cdots,n_t\right)}.$$

En estadística, el coeficiente de determinación es la proporción de la varianza de la variable dependiente F_i que es predecible a partir de la variable independiente $F(i)$. Es un estadístico utilizado en el contexto de la bondad del ajuste y proporciona una medida de lo bien que los resultados observados son replicados por el modelo, basándose en la proporción de la variación total de los resultados explicada por el modelo. Este coeficiente toma valores entre 0 y 1, y entre dos modelos se prefiere el que tenga el coeficiente de determinación más alto. Además, con este criterio, el mejor modelo es el que maximiza el coeficiente de determinación dentro de una familia plausible de modelos:

$$\max_{n,\mu,\sigma} R^2\left(t,n,\mu,\sigma|n_1,\cdots,n_t\right)\ .$$

Por otro lado, vamos a controlar al mismo tiempo el ajuste de las frecuencias observadas mediante la función de densidad teórica. Para ello, el criterio que vamos a seguir es realizar una regresión lineal

$$\frac{f_i}{f\left(i\right)}=a+bi+e_i, e_i\sim N\left(0,\nu\right), i=1,\cdots,t,$$

y al introducir la restricción

$$p-value\left(F_{obs}\right)=P\left(F_{1,t-2}>F_{obs}\right)>0.1,$$

donde F_{obs} es el valor observado del estadístico de prueba para comprobar testing $H_0: b=0 \ \ vs\ H_1: b\neq 0$ y $F_{1,t-2}$ es su distribución teórica bajo H_0, que es una distribución F de Snedecor con 1 y $t-2$ grados de libertad.

Proponemos resolver el problema de optimización multicriterio obteniendo los valores de $n\left(t\right)$, $\mu\left(t\right)$ y $\sigma\left(t\right)$, de modo que

$$\max_{n,\mu,\sigma} R^2\left(t,n,\mu,\sigma|n_1,\cdots,n_t\right)\ ,$$

bajo la restricción

$$p-value\left(F_{obs}\right)=P\left(F_{1,t-2}>F_{obs}\right)>0.1.$$

Ahora, stop si $F\left(t\right)=F_t=0.5$, de lo contrario incorpore el dato t_1, haga $t=t_1$ y repita.

Paralelamente, se ajusta un modelo para cada una de las series: número de nuevos casos positivos por día, número de nuevas muertes por día y número de nuevas UCI por día, y se elije el modelo que maximice simultáneamente los tres valores de R^2.

paramos cuando $F\left(t\right)=F_t=0.5$ cuando se consigue en las tres series.

Es importante destacar que el algoritmo permite la introducción de nuevos datos en tiempo real y es escalable.

B. La Heurística

Para realizar una optimización efectiva, hemos optado por una ambiciosa heurística que detallamos a continuación.

Sea $\sigma=\sigma_0$, empezando en $\sigma_0=15$.

Variando σ entre σ_0-14 y σ_0+14.

En este punto es importante tener en cuenta que el periodo de incubación de la enfermedad oscila entre 2 y 14 días (véase [14]). Además, se considera el retraso entre el momento de la infección y la notificación como caso positivo.

Sea $\mu=\mu_0$, empezando en $\mu_0=t_0$, t_0 es el momento presente.

Se mueve μ entre el primer día de cada una de las series y $t_0+2\sigma_0$. Por ejemplo, el primer día de la serie de número de infectados en la Comunidad de Madrid es el día 26, que corresponde al 25 de febrero.

Generar $k=10000$ valores de una variable aleatoria uniforme entre 0 y 1.

Calcular $n=Np$ para cada valor p generado en el último paso, $N=6550000$ aproximadamente en la Comunidad de Madrid.

Descartar los valores de $n<N_{t_0}$.

Encontrar los modelos factibles con valor $p>0,1$ para el ruido y seleccionar el que tenga el mayor R^2 de ajuste en las frecuencias acumuladas.

En la práctica, la ejecución de la heurística genera un archivo .csv que contiene varias columnas. Se incluyen las columnas correspondientes a los parámetros ajustados, μ, σ y n, el coeficiente de determinación y el valor p. Además, se añaden dos columnas para registrar cada día el momento del pico real, que se corresponde con el día de mayor frecuencia observado hasta la fecha, y el día en el que se observa el corte entre los modelos ajustados a ambos lados, es decir, cuando la distribución se vuelve positivamente sesgada. El algoritmo trata de hacer coincidir el valor del pico real, el corte y el parámetro μ. También permite ajustar un σ diferente a la izquierda y a la derecha del corte. Las dos últimas columnas incluyen las fechas de compromiso correspondientes a los percentiles $qnorm_{0.99}$ y $qnorm_{0.999}$ del modelo ajustado a la derecha.

III. Modelo de Correción del error

Es intuitivo pensar que un pico en la serie confirmada aumentará las otras tres series de datos con cierto retraso. Este tipo de relación se suele modelar incluyendo el desplazamiento de la serie confirmada p veces hacia el futuro y utilizando esta característica para predecir el presente de otras series. Este método para una serie se denomina modelo autorregresivo y se generaliza a más de una serie en el modelo autorregresivo vectorial. Este tipo de modelo necesita que la serie temporal sea estacionaria, lo que significa que el proceso tiene los dos primeros momentos constantes. Esta restricción es un problema en este caso, en el que queremos predecir los cambios de tendencia futuros.

Cuando se tiene un conjunto de series temporales estacionarias, $y_t=(y_{1t},...,y_{Kt})$, se puede definir el modelo vectorial autorregresivo estable de orden p como

$$y_t = \nu + A_1 y_{t-1} + \ldots + A_p y_{t-p} + u_t$$

donde los A_i son matrices de parámetros, $\nu = (\nu_1, ..., \nu_K)$ representa la media de cada serie temporal y u_t representa el error del modelo, donde $u_t \sim \mathcal{N}(0, \Sigma_u)$.

El procedimiento habitual cuando se trabaja con este modelo con series no estacionarias es diferenciarlas hasta que se vuelvan estacionarias, pero este procedimiento enturbia la inferencia sobre el modelo.

Habrá una relación a largo plazo. La tendencia estocástica de las series será común ya que bajarán y subirán de la misma manera, guardando cierta distancia entre ellas.

Podemos decir que el conjunto de series temporales y_t tienen un equilibrio largo si existe un vector β tal que $\beta^{'} y_t = \beta_1 y_{1t} + ... + \beta_K y_{Kt} = 0$ y definir el proceso $z_t = \beta^{'} y_t$ como las desviaciones de esta relación.

Así, las series del conjunto y_t se dicen cointegradas si y_{it} es una serie de orden 1 para todo $i = 1, ..., K$ y existe un vector β tal que $z_t = \beta^{'} y_t$ es un proceso estacionario.

Teniendo en cuenta esta relación, es posible definir un modelo basado en el vector autorregresivo para este tipo de datos, se trata del modelo de corrección del error y viene dado por

$$\begin{aligned}\Delta y_t &= \Pi y_{t-1} + \Gamma_1 \Delta y_{t-1} + \ldots + \Gamma_{p-1} \Delta y_{t-p+1} + \phi d_t + u_t \\ &= \alpha\beta' y_{t-1} + \Gamma_1 \Delta y_{t-1} + \ldots + \\ &\quad \Gamma_{p-1} \Delta y_{t-p+1} + \phi d_t + u_t \quad (1)\end{aligned}$$

donde Γ_i son matrices de parámetros, d_t es la parte determinista del modelo,
$u_t \sim \mathcal{N}(0, \Sigma_u)$ y $\Pi = \alpha\beta^{'}$ es la matriz de carga y β la matriz de cointegración.

$$\begin{aligned}\Delta y_t &= \Pi_1 y_{t-1} + \Gamma_{1,1} \Delta y_{t-1} + \ldots + \Gamma_{1,p-1} \Delta y_{t-p+1} \\ &\quad + \phi_1 d_t + u_t = \alpha_1 \beta_1' y_{t-1} + \Gamma_{1,1} \Delta y_{t-1} \\ &\quad + \ldots + \Gamma_{1,p-1} \Delta y_{t-p+1} + \phi_1 d_t + u_t \text{ for } t \leq T_1 \quad (2)\end{aligned}$$

$$\begin{aligned}\Delta y_t &= \Pi_2 y_{t-1} + \Gamma_{2,1} \Delta y_{t-1} + \ldots + \Gamma_{2,p-1} \Delta y_{t-p+1} \\ &\quad + \phi_2 d_t + u_t = \alpha_2 \beta_2' y_{t-1} + \Gamma_{2,1} \Delta y_{t-1} \\ &\quad + \ldots + \Gamma_{2,p-1} \Delta y_{t-p+1} + \phi_2 d_t + u_t \text{ for } t > T_1 \quad (3)\end{aligned}$$

Una vez definido el modelo, sólo hay que decidir un parámetro, el orden de regresión p, es decir, el número de desplazamientos de cada una de las series de la ecuación. Para la decisión de este parámetro y dada la facilidad de cálculo de este modelo, se realiza un procedimiento de validación cruzada con diferentes p, tomando el orden p con menor error porcentual absoluto medio.

Para montar el modelo se procede como se ha descrito previamente.

Se aplica una transformación logarítmica a los datos y se comprueba la cointegración dos a dos de las series de estudio con el procedimiento de Engle y Granger [15] y el test de Phillips-Ouliaris [15].

Test	Combination	p-value
Phillips Ouliaris	conf - hosp	< 0.01
Engle and Granger	conf - hosp	< 0.01
Phillips Ouliaris	conf - UCI	< 0.01
Engle and Granger	conf - UCI	< 0.01
Phillips Ouliaris	conf - fallecidos	< 0.01
Engle and Granger	conf - fallecidos	< 0.01
Phillips Ouliaris	hosp - UCI	< 0.01
Engle and Granger	hosp - UCI	< 0.01
Phillips Ouliaris	hosp - fallecidos	< 0.01
Engle and Granger	hosp - fallecidos	< 0.01
Phillips Ouliaris	UCI - fallecidos	< 0.01
Engle and Granger	UCI - fallecidos	< 0.01

TABLE I: Tests de Cointegración para la Comunidad de Madrid

IV. Modelo SIR con confinamiento: Modelo SCIR

Para las comparaciones entre modelos, hemos considerado la siguiente extensión del modelo SIR propuesto por [6]. Hemos elegido este modelo porque incorpora el compartimento de las muertes en su definición y se formula sólo a través de cinco parámetros. Se pueden encontrar diferentes extensiones de los modelos SIR en [16] y [17].

El modelo SCIR incluye los estados habituales de un modelo SIR más una clase C para los individuos enviados al confinamiento que son susceptibles pero no infectados. Los individuos susceptibles (S) pueden entrar y salir del confinamiento (C) o infectarse (I). Los individuos infectados pueden recuperarse (R) o morir (D). La figura 1 muestra el diagrama de las ecuaciones del modelo SCIR.

Para la optimización de los parámetros, hemos incluido en la función objetivo una medida de precisión que combina los coeficientes de determinación de los ajustes tanto de los casos como de las muertes.

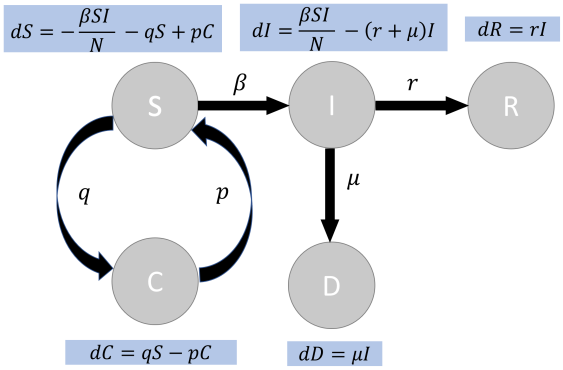


Fig. 1: Diagrama del Modelo SCIR [6]

V. Resultados Experimentales

El pico real de la serie es uno de los valores más difíciles de predecir. Algunos días después de que se haya producido,

es fácil saber que el pico se ha alcanzado. La figura 3 indica que μ coincide con el valor del pico real entre el 17 y el 21 de abril. Sin embargo, como la curva de variación de μ por día (ver Figura 3) comienza a aplanarse a partir del 20 de marzo, el modelo nos alerta de antemano de la posibilidad de que el pico real aparezca en cualquier momento después de esa fecha.

En una situación de transmisión sin virus, el modelo se ajustaría a una distribución gaussiana perfecta y μ sería igual al pico real. Por lo tanto, las pequeñas desviaciones del modelo a la izquierda del pico real pueden indicar un cambio en la evolución del virus. Por ejemplo, entre el 8 y el 14 de marzo, la curva de las frecuencias relativas (azul) está por encima del modelo ajustado para la función de densidad (naranja) (véase la figura 2), lo que puede indicar la situación de peligro presente en Madrid antes del 8 de marzo. Esta situación de peligro podría ser consecuencia de los sucesos puntuales que aumentan el contacto estrecho entre las personas. Esto ya era evidente en los modelos ajustados hasta esta fecha. El periodo anterior al 11 de marzo puede considerarse libre de transmisión de la enfermedad porque no se han aplicado intervenciones. Sin embargo, ese modelo libre de virus se observa varios días después de esa fecha. De hecho, el 17 de marzo se produce un pequeño pico de casos, que podría deberse a un elevado número de contagios durante diferentes eventos masivos en Madrid el 8 de marzo. El 22 de marzo se empiezan a notar las medidas impuestas por el gobierno, ya que los datos observados están por debajo del modelo ajustado, y esa situación se mantiene hasta que se alcanza el pico global de casos alrededor del 26 de marzo. El 30 de marzo, tras el pico de casos, la libertad de actividad fuera del hogar se redujo a los servicios esenciales. El efecto de esta intervención se nota el 9 de abril, cuando los datos reales están por debajo del modelo ajustado, y esa tendencia se mantiene hasta el 12 de abril. Parece que las intervenciones tardan unos 11 días en notarse.

Es importante señalar que el parámetro σ del modelo cambia a la derecha del pico real de casos. Esto indica que las medidas de contención son efectivamente eficaces. Esto se traduce en un aumento de la varianza del modelo a la derecha del punto de inflexión μ, lo que indica una ralentización de las infecciones (véase la figura 3). El 18 de abril, con la adición de la columna TestAc al conjunto de datos, se observa una explosión en el gráfico de la densidad de casos (véase la figura 2). Salvo este incidente, el modelo se mantiene bastante estable a la derecha del pico de casos y se pueden proponer fechas de compromiso para el levantamiento progresivo de las restricciones de movilidad, como muestra la Figura 3. Por ejemplo, el 11 de mayo muestra fechas de compromiso entre el 19 de mayo y el 5 de junio sobre la base de 0,01 y 0,001 para la zona de cola derecha del modelo (con una previsión de 72200 para el total de casos confirmados al final de la pandemia).

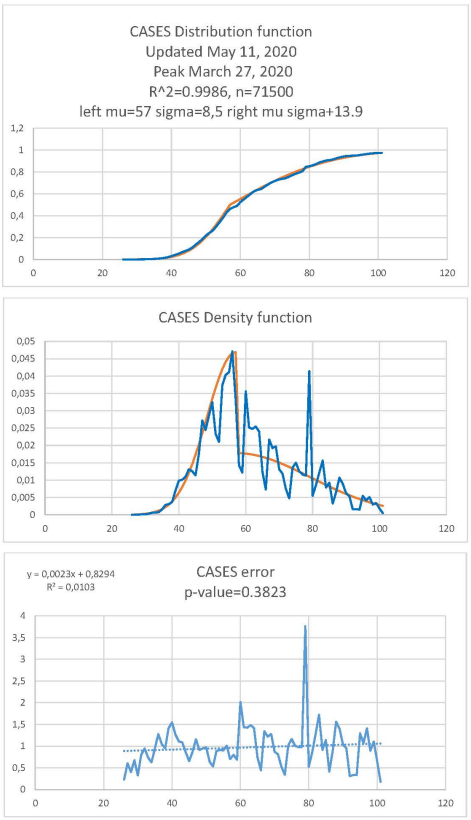


Fig. 2: Informe de CASOS generado automáticamente por nuestro sistema experto. De arriba a abajo: ajuste de la función de distribución, ajuste de la función de densidad y ruido el 11 de mayo.

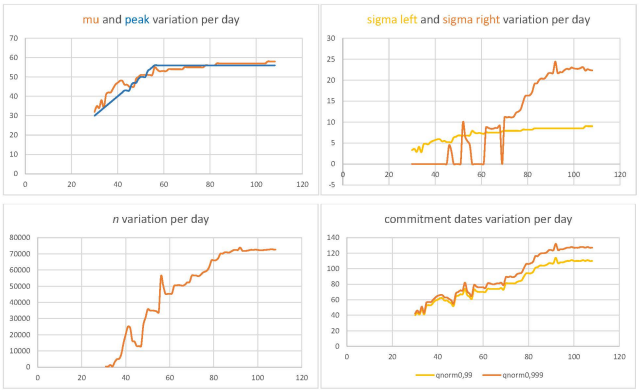


Fig. 3: Variación de los parámetros por día hasta el 11 de mayo: este informe ha sido generado automáticamente por nuestro sistema experto. De arriba a abajo: pico, sigma izquierda y sigma derecha, y n

VI. Conclusiones

Se desarrolla e implementa un modelo de regresión no lineal para datos de conteo en series temporales mediante un sistema experto de Inteligencia Artificial. Se basa en la estimación directa de la función de distribución de cada una de las series en estudio y en la dualidad entre la función de distribución y la función de densidad. Dado que estas dos funciones caracterizan completamente la distribución de probabilidad de una variable, nuestro modelo es capaz de captar las principales características de los brotes epidémicos. También hay que destacar la simplicidad de MATGEN, ya que se formula sólo a través de cuatro parámetros, μ, σ_{left}, σ_{right} y n. El seguimiento de todos los parámetros del modelo permite cuantificar y detectar fácilmente el efecto de las intervenciones a lo largo del tiempo. Además, el algoritmo de Machine Learning desarrollado es escalable, permite la ejecución en paralelo de diferentes series de datos y es capaz de introducir nuevos datos en tiempo real.

Aplicamos este modelo a la serie COVID-19 de la Comunidad de Madrid (España) durante la primera y segunda oleada para dar una previsión a 8 días de la iniciativa matemática española [11]. Este marco teórico permite detectar los picos pandémicos y hacer un seguimiento y previsión a corto y largo plazo del número de infectados, personas que requieren hospitalización y fallecidos. Este sistema experto resulta muy útil para conocer la eficacia de las intervenciones impulsadas por el gobierno, que parecen tener un impacto después de 11 días de su aplicación durante la primera ola. Además, es útil para proponer fechas de compromiso para levantar las restricciones de movilidad y para aconsejar cómo proceder en futuros brotes. El 25 de mayo, la Comunidad de Madrid entró en la fase 1 de la desescalada. La actualización de MATGEN del 11 de mayo mostró fechas de compromiso entre el 19 de mayo y el 5 de junio (con una previsión de 72200, 9000 y 4000 para el total de casos confirmados, muertes y UCIs al final de la pandemia respectivamente).

Finalmente, la comparación desde distintos puntos de vista de nuestros dos modelos con el modelo SCIR [6] arroja las siguientes conclusiones:

- Entre las ventajas de utilizar el Modelo SCIR están su sencillez al estar formulado con cinco parámetros, y la interpretabilidad de los mismos desde el punto de vista epidemiológico.
- El Modelo de Regresión No Lineal MATGEN (formulado con cuatro parámetros por serie que son fáciles de controlar) es el más explicativo para estudiar la detección de picos y el efecto de las intervenciones. Además, está dotado de un procedimiento de control que permite detectar los cambios de tendencia en las colas que indican el inicio de una nueva ola. A diferencia de los otros dos modelos, el ajuste de las cuatro series es en paralelo.
- El Modelo de Corrección de Errores MATGEN es el que depende de un mayor número de parámetros, pero permite aproximar las cuatro series simultáneamente y es el mejor en todas las métricas, tanto en ajuste como en previsión. Además, este modelo incorpora la función de respuesta al impulso, un método que permite hacer inferencia sobre el impacto de una serie sobre las demás.

Por lo tanto, MATGEN combina nuestros dos modelos propuestos en un sistema experto como una nueva herramienta epidemiológica que puede resultar extremadamente útil en nuevos brotes de COVID-19 y futuras epidemias de enfermedades infecciosas.

Acknowledgment

Trabajo financiado por Comunidad de Madrid y de los fondos FEDER a través del proyecto S2017/BMD3773 (GenObIA-CM) y del proyecto de Doctorado Industrial IND2020/TIC-17435, y por el Ministerio de Economía y Competitividad bajo el proyecto RTI2018-095180-B-I00.

Bibliografía

[1] W. Naudé. Artificial Intelligence Against Covid-19: An Early Review. *IZA Discussion Paper*, 13110, 2020. Available at SSRN https://ssrn.com/abstract=3568314.

[2] F. Verelst, L. Willem, and F. Beutels. Behavioural change models for infectious disease transmission: a systematic review (2010-2015). *Journal of the Royal Society Interface*, e 13(20160820), 2016. https://doi.org/10.1098/rsif.2016.0820.

[3] B. Ivorra, M. R. Ferrández, M. Vela-Pérez, and A. M. Ramos. Mathematical modeling of the spread of the coronavirus disease 2019 (covid-19) taking into account the undetected infections. the case of china. *Communications in nonlinear science & numerical simulation*, 88:105303–105303, 2020.

[4] L. Wang, Y. Zhou, J. He, B. Zhu, F. Wang, L. Tang, M. Eisenberg, and P. X. K. Song. An epidemiological forecast model and software assessing interventions on covid-19 epidemic in china. *medRxiv*, 2020. preprint on webpage at https://doi.org/10.1101/2020.02.29.20029421.t.

[5] B. F. Maier and D. Brockmann. Effective containment explains subexponential growth in confirmed cases of recent COVID-19 outbreak in mainland china. *medRxiv*, 2020. preprint on webpage at https://doi.org/10.1101/2020.02.18.20024414.

[6] Mario Castro, Saúl Ares, José A. Cuesta, and Susanna Manrubia. The turning point and end of an expanding epidemic cannot be precisely forecast. *Proceedings of the National Academy of Sciences*, 117(42):26190–26196, 2020.

[7] Pablo Sánchez-Villegas and Antonio Daponte Codina. Predictive models of the covid-19 epidemic in spain with gompertz curves. *Gaceta sanitaria*, pages S0213–9111(20)30125–4, 2020.

[8] J. A. Nelder and R. Mead. A Simplex Method for Function Minimization. *Comput. J.*, 7:308–313, 1965.

[9] John C. Nash. *Compact numerical methods for computers : linear algebra and function minimisation*. Hilger, Bristol, Eng.; New York, 1990.

[10] Martí Català, Sergio Alonso, Enrique Alvarez-Lacalle, Daniel López, Pere-Joan Cardona, and Clara Prats. Empiric model for short-time prediction of covid-19 spreading. *medRxiv*, 2020.

[11] CEMAT. Cooperative prediction. https://covid19.citic.udc.es/, 2021.

[12] ISCIII. Instituto de Salud Carlos III. https://cnecovid.isciii.es/covid19, 2020.

[13] M. A. Gómez Villegas. *Inferencia Estadística*. Díaz de Santos, Spain, 2011.

[14] S. A. Lauer, K. H. Grantz, Q. Bi, F. K. Jones, Q. Zheng, H. R. Meredith, A. S. Azman, N. G. Reich, and J. Lessler. The incubation period of coronavirus disease 2019 (covid-19) from publicly reported confirmed cases: Estimation and application. *Ann Intern Med*, 172(9):577–582, 2020.

[15] Adrian Trapletti and Kurt Hornik. *tseries: Time Series Analysis and Computational Finance*, 2019. R package version 0.10-47.

[16] Md. Sharif Uddin, Md. Taufiq Nasseef, Mufti Mahmud, and Ali AlArjani. Mathematical modelling in prediction of novel coronavirus (covid-19) transmission dynamics. 9 2020.

[17] Fernando Pazos and Flavia E. Felicioni. A control approach to the covid-19 disease using a seihrd dynamical model. *medRxiv*, 2020.

Calibración de un modelo basado en agentes que describe la evolución de la resistencia del *Acinetobacter baumannii* a la colistina en la ciudad de Valencia

Juan A. Aledo
Departamento de Matemáticas
Universidad de Castilla-La Mancha
Albacete, España
juanangel.aledo@uclm.es

Carlos Andreu-Vilarroig
Instituto Universitario de Matemática Multidisciplinar
Universitat Politècnica de València
Valencia, España
caranvi1@upv.es

Juan-C. Cortés
Instituto Universitario de Matemática Multidisciplinar
Universitat Politècnica de València
Valencia, España
jccortes@imm.upv.es

Juan C. Orengo
Public Health Program
Ponce Health Sciences University and Merck & Co, Inc.
Ponce, Puerto Rico
juan_orengo@merck.com

Rafael-J. Villanueva
Instituto Universitario de Matemática Multidisciplinar
Universitat Politècnica de València
Valencia, España
rjvillan@imm.upv.es

***Abstract*—Uno de los retos más complejos en el mundo de la sanidad es la resistencia a los tratamientos con antibióticos (AMR). El uso a nivel mundial de los fármacos antibióticos ha sido un factor clave en la lucha contra las enfermedades infecciosas, pero a su vez ha provocado que cada vez más microorganismos desarrollen, mediante adaptaciones genéticas, resistencia o incluso inmunidad frente a los fármacos. Ante esta amenaza de salud pública, los modelos matemáticos van a resultar fundamentales en la predicción de la evolución de la AMR y en la toma de decisiones objetivas en materia de salud pública. Por ello, en este trabajo proponemos un modelo basado en agentes que describe la evolución de la bacteria *Acinetobacter baumannii* resistente al antibiótico colistina en una población sintética con las características demográficas de la ciudad de Valencia. Para hallar los parámetros del modelo y simular un escenario realista, se ha llevado a cabo un proceso de calibrado, utilizado para ello datos epidemiológicos de la ciudad de Valencia y de España. Para la exploración del espacio de parámetros, se ha empleado el algoritmo de optimización *Particle Swarm Optimization* (PSO) y una función de *fitness* que mide el error entre los resultados del modelo y los datos epidemiológicos reales.**

***Palabras clave*—Resistencia antimicrobiana (AMR), *Acinetobacter baumannii*, Colistina, Modelo basado en agentes, Calibración, Optimización**

Esta investigación ha sido financiada por el Ministerio de Economía, Industria y Competitividad (MINECO), la Agencia Estatal de Investigación (AEI) y Fondo Europeo de Desarrollo Regional (FEDER UE) MTM2017-89664-PI, y el Instituto Universitario de Matemática Multidisciplinar de la Universitat Politècnica de València.

I. Introducción

Uno de los retos emergentes en el mundo de la salud es la resistencia antimicrobiana (*antimicrobial resistance* o AMR, por sus siglas) a los tratamientos antibióticos. Los tratamientos antibióticos son aquellos agentes que actúan sobre microorganismos bacterianos. Sin embargo, dichos microorganismos son capaces de desarrollar resistencia al medicamento mediante adaptaciones genéticas, y generar variantes resistentes o incluso inmunes al tratamiento.

Aunque en los últimos años la Organización Mundial de la Salud (OMS), la Asamblea General de Naciones Unidas y otras organizaciones han declarado a la AMR como una de las diez principales amenazas globales de salud pública contra la humanidad [1], lo cierto es que las advertencias sobre la AMR y las consecuencias de la inacción ante este problema se han sucedido durante los últimos cincuenta años. En numerosos estudios científicos de impacto, los investigadores alertan de que la prescripción de medicamentos antibióticos sin control, el exceso en el consumo de antibióticos en el sector agroalimentario y la falta de políticas regulatorias generan una enorme presión selectiva que contribuye a aumentar la AMR rápidamente [2]–[4].

Por otro lado, dentro del campo de la epidemiología, el modelado matemático ha resultado ser un gran avance para la predicción y control de la dinámica de las enfermedades en

grandes poblaciones, la toma de decisiones de salud pública (como los programas de vacunación) o su impacto económico, entre otros. Sin embargo, son pocos los modelos matemáticos que incorporan el factor de la AMR, debido a limitaciones como la falta de datos clínicos, su incompletitud, o la gran heterogeneidad de las fuentes. Además, debido a su complejidad, los modelos que estudian la AMR se suelen focalizar en una serie de patógenos concretos y en la transmisión de las enfermedades que provocan.

En este trabajo se propone el calibrado de un modelo basado en agentes para estudiar el comportamiento resistente de la bacteria *A. baumannii* con respecto al antibiótico colistina (generalmente empleado como último recurso por sus nocivos efectos secundarios) y estimar el tiempo que se va a poder utilizar este antibiótico hasta que la bacteria se vuelva mayoritariamente resistente en la población.

II. Construcción del modelo

Un modelo basado en agentes es un tipo de modelo computacional que presenta dos elementos fundamentales: *(i)* una *población sintética* constituida por individuos con unas características y *(ii)* unas reglas de comportamiento o *dinámica* poblacional que rige la evolución de los individuos.

A. Población sintética

Para generar la lista de individuos o agentes que conforman la población sintética de la ciudad de Valencia, tomamos los datos demográficos del número de habitantes y la distribución de la población por grupos quinquenales de edad de la ciudad durante los años 2012-2020, disponibles en el Portal Estadístico de la Generalitat Valenciana (GVA) [5]

En referencia a la población total (en número de habitantes), se observa que la distribución temporal de la población durante los años 2012-2020. sigue una distribución uniforme discreta (test χ^2, p-valor $\simeq 1$). Por lo tanto, tomamos la población media de la ciudad de Valencia durante dicho periodo - $N = 782\,750$ - como el tamaño fijo de la población sintética. Una vez establecido el tamaño de la población, se le asigna a cada individuo un grupo quinquenal de edad aleatorio $G \sim \hat{p}_G$, que sigue la distribución empírica $\hat{p}_G$ reflejada en la Fig. 1. La asignación de la edad E de los individuos, dado un grupo de edad G, se realiza de forma aleatoria uniforme: $E|G \sim \text{Unif}\left(\left[e_i^G, e_s^G\right]\right)$, donde e_i^G y e_s^G son las edades inferior y superior que acotan al grupo de edad G, respectivamente.

B. Dinámica

Para la definición de la dinámica del modelo, además de la edad, es necesario definir en cada individuo de la población una serie de estados adicionales:

1) *Entorno* en el que se encuentra el individuo: comunidad (C) u hospital (H).
2) *Estado de salud* del individuo con respecto a la enfermedad por *A. baumannii*: susceptibles (X), sensibles (S) y resistentes (R).
3) *Origen de infección*: no presentan infección ($-$), infección comunitaria (IC) o infección nosocomial (IN).

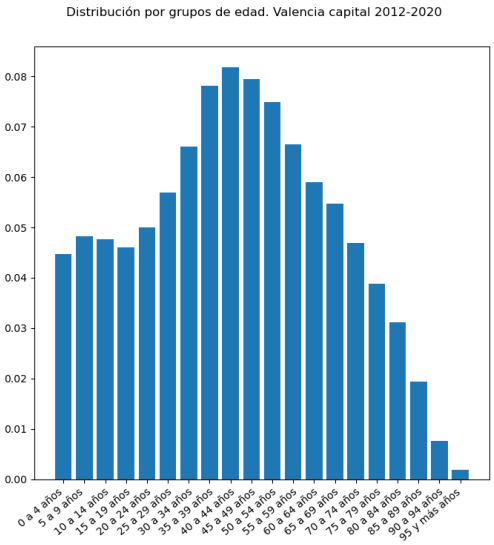


Fig. 1. Distribución por grupos de edad de la población de la ciudad de Valencia en el periodo promedio 2012-2020.

4) *Tiempo de hospitalización* T_h. En caso de encontrarse en la comunidad, establecemos que el tiempo de hospitalización sea $T_h = 0$.

De este modo, un individuo de la población sintética queda caracterizado por un vector de cinco estados.

La dinámica del modelo sigue el esquema del proceso representado en la Fig. 2, y toma como paso temporal 1 día. Para la explicación de la dinámica en el esquema tan solo se consideran el *estado de salud* y *origen de la infección* para mayor claridad. La dinámica para cualquier individuo de la población es la siguiente:

I) Un individuo $(X, -)$ situado en la comunidad C ingresa en el hospital H en un instante t con una probabilidad $\hat{p}_{H|G}$ y se le asigna un tiempo de estancia hospitalaria $T_h|G$ en función de su grupo de edad G. Allí, tras las pruebas pertinentes, se le puede diagnosticar que:
   a) $(X, -) \to (S, IC)$ ingresa por una infección comunitaria por *A. baumannii* sensible a colistina con una probabilidad $\alpha_1 = \pi_{IC|CH}\ \pi_{IAB|IC}\ (1 - \hat{p}_{R|IAB})$, donde $\pi_{IC|CH}$, $\pi_{IAB|IC}$ y $\hat{p}_{R|IAB}$ son las probabilidades estimadas de infección comunitaria dado un ingreso hospitalario, de infección por *A. baumannii* dada una infección comunitaria, y de infección por una cepa resistente a colistina dada una infección por *A. Baumannii*, respectivamente,
   b) $(X, -) \to (R, IC)$ ingresa por una infección comunitaria por *A. baumannii* resistente a colistina con una probabilidad $\alpha_2 = \pi_{IC|CH}\ \pi_{IAB|IC}\ \hat{p}_{R|IAB}$,
   c) $(X, -) \to (X, IC)$ ingresa por una infección comunitaria no provocada por *A. baumannii* con una probabilidad $\alpha_3 = \pi_{IC|CH}\ (1 - \pi_{IAB|IC}) = \pi_{IC|CH} - \alpha_1 - \alpha_2$,
   d) $(X, -) \to (X, -)$ ingresa por otras causas no infecciosas con una probabilidad $\alpha_4 = 1 - \pi_{IC|H} =$

$1 - \alpha_1 - \alpha_2 - \alpha_3$.

II) Un individuo ingresado en el hospital H en el instante t por causas no infecciosas $(X, -)$, durante su estancia hospitalaria puede:
   a) $(X, -) \rightarrow (S, IN)$ contraer una infección nosocomial por *A. baumannii* sensible a colistina con una probabilidad $\beta_1 = \pi_{IN|H}\ \pi_{IAB|IN}\ (1 - \hat{p}_{R|IAB})$, donde $\pi_{IN|H}$ es un estimador de la probabilidad de contraer una infección nosocomial dada una estancia hospitalaria, $\pi_{IAB|IN}$ es un estimador de la probabilidad de infección por *A. baumannii* dada una infección nosocomial,
   b) $(X, -) \rightarrow (R, IN)$ contraer una infección nosocomial por *A. baumannii* resistente a colistina con una probabilidad $\beta_2 = \pi_{IN|H}\ \pi_{IAB|IN}\ \hat{p}_{R|IAB}$,
   c) $(X, -) \rightarrow (X, IN)$ contraer una infección nosocomial no provocada por *A. baumannii* con una probabilidad $\beta_3 = \pi_{IN|H}\ (1 - \pi_{IAB|IN}) = \pi_{IN|H} - \beta_1 - \beta_2$,
   d) $(X, -) \rightarrow (X, -)$ mantenerse susceptible sin ningún tipo de infección con una probabilidad $\beta_4 = 1 - \pi_{IN|H} = 1 - \beta_1 - \beta_2 - \beta_3$.

III) Un individuo ingresado en el hospital H ha ingresado en el instante t y ha contraído durante su estancia hospitalaria una infección nosocomial (X, IN), durante el resto de su estancia puede:
   a) $(X, IN) \rightarrow (S, IN)$ contraer una infección nosocomial por *A. baumannii* sensible a colistina con una probabilidad $\gamma_1 = \beta_1 = \pi_{IN|H}\ \pi_{IAB|IN}\ (1 - \hat{p}_{R|IAB})$,
   b) $(X, IN) \rightarrow (R, IN)$ contraer una infección nosocomial por *A. baumannii* resistente a colistina con una probabilidad $\gamma_2 = \beta_2 = \pi_{IN|H}\ \pi_{IAB|IN}\ \hat{p}_{R|IAB}$,
   c) $(X, IN) \rightarrow (X, IN)$ mantenerse susceptible con una infección nosocomial no provocada por *A. baumannii* con una probabilidad $\gamma_3 = 1 - \beta_1 - \beta_2 = 1 - \pi_{IN|H}\ \pi_{IAB|IN}$.

IV) Cualquier individuo situado en el entorno hospitalario H sale de nuevo a la comunidad C al finalizar su estancia hospitalaria, es decir, cuando se cumplen T_h días de estancia.

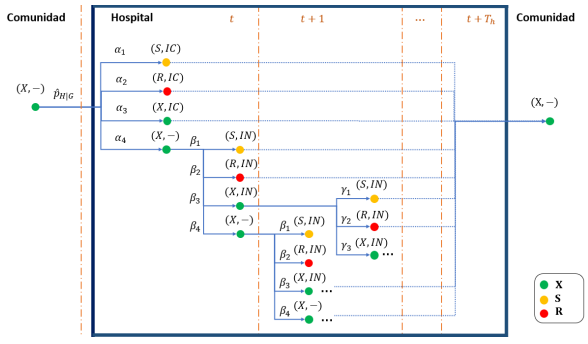


Fig. 2. Esquema de la dinámica del modelo basado en agentes.

C. Parámetros

De los parámetros descritos en la dinámica, la distribución empírica de probabilidad de ingreso dado el grupo de edad $\hat{p}_{H|G} \simeq \mathbb{P}(H|G)$ (Fig. 3) se ha deducido a partir de los datos de la provincia de Valencia entre los años 2016-2019, disponibles en el Portal Estadístico del Ministerio de Sanidad [6].

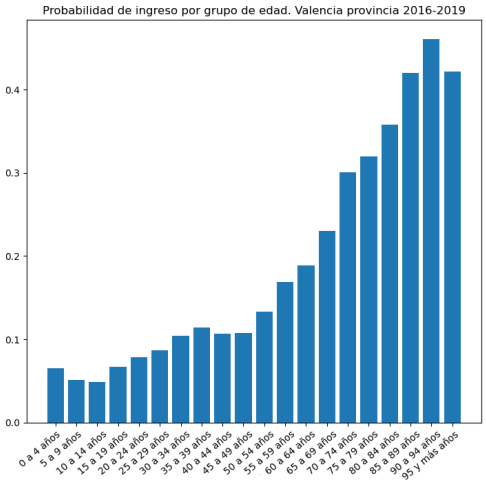


Fig. 3. Distribución empírica de la probabilidad $\hat{p}_{G|H}$ anual de ingreso dado el grupo de edad de Valencia provincia en el periodo promedio 2016-2019. Observamos que la probabilidad de ingreso es más elevada en los grupos de mayor edad.

De la misma fuente y en el mismo periodo se han obtenido los tiempos de estancia media hospitalaria por grupo de edad $\bar{T}_h|G$ para la provincia de Valencia y los años 2016-2019 (Fig. 4), y que son un estimador del valor esperado del tiempo de estancia aleatorio $T_h|G$. Al no existir variaciones estadísticamente significativas entre años, se ha tomado el promedio para cada grupo de edad. En este modelo, establecemos que $T_h|G = \bar{T}_h|G + \delta\bar{T}_h|G = (1 + \delta)\bar{T}_h|G$, donde el factor δ representa un incremento porcentual en el tiempo de hospitalización en función del estado de salud. Según [7], los pacientes con cepa sensible S incrementan la estancia media en un 6.45% ($\delta = 0.0645$), mientras que los pacientes con cepa resistente R incrementan la estancia media en un 38.89% ($\delta = 0.3889$). Los pacientes susceptibles X no incrementan su estancia media ($\delta = 0$).

Por otro lado, el estimador $\hat{p}_{R|IAB}$ de la probabilidad de resistencia a la colistina dada una infección por *A. baumannii* se ha extraído de los informes anuales de la Red de Vigilancia Microbiológica de la Comunidad Valenciana (REDMIVA) entre los años 2015 y 2018 [8]. En los datos reflejados en la Tabla I, observamos una tendencia creciente en la proporción de pacientes resistentes, que es coherente con el proceso de adaptación de las bacterias a los tratamientos antibióticos.

Para modelar el crecimiento de la resistencia a antibiótico se propone una función sigmoide $\sigma(t)$, con rango en el intervalo $[0, 1]$ [9]:

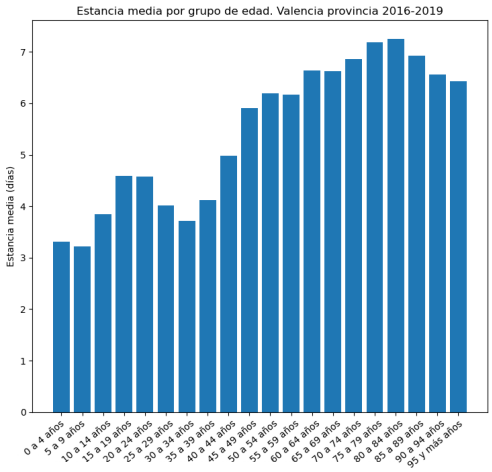


Fig. 4. Tiempo de estancia media hospitalaria $\bar{T}_h|E$ por grupo de edad en el periodo promedio 2016-2019 en la provincia de Valencia.

TABLA I
TAMAÑOS MUESTRALES Y PROPORCIONES DE RESISTENCIA DEL *A. baumannii* A LA COLISTINA. COMUNIDAD VALENCIANA, PERIODO 2015-2018.

Año	**Tamaño muestral** n_{IAB}	$\hat{p}_{R\|IAB}$
2015	505	0.0139
2016	376	0.0426
2017	406	0.0911
2018	594	0.0976

$$\hat{p}_{R|IAB} = \sigma(t) = \frac{1}{1 + e^{-K(t-t_0)}}, \quad (1)$$

$$\hat{p}_{S|IAB} = 1 - \sigma(t), \quad (2)$$

donde K es la tasa de crecimiento exponencial de la resistencia a antibiótico y t_0 es el tiempo en el que la probabilidad de resistencia alcanza el 50%. Para hallar los parámetros óptimos K y t_0, se ha aplicado un ajuste con los datos anuales de los informes REDMIVA. El resultado gráfico del ajuste se muestra en la Fig. 5, y los parámetros óptimos, en la Tabla II. Tal y como se observa en la gráfica, actualmente la resistencia a la colistina se encontraría en las primeras fases.

TABLA II
PARÁMETROS ÓPTIMOS DE LA FUNCIÓN DE PROBABILIDAD SIGMOIDE $\sigma(t)$ AJUSTADA A LOS DATOS.

Parámetros	
K (días^{-1})	t_0 (días)
1.287×10^{-3}	3.109×10^{3}

En cuanto a las probabilidades de infección intrahospitalarias, $\pi = \{\pi_{IC|CH}, \pi_{IN|H}, \pi_{IAB|IC}, \pi_{IAB|IN}\}$, son probabilidades desconocidas de las cuales no se tienen estimadores. Sin embargo, dado el modelo, es posible estimarlas indirectamente con datos de prevalencias. El Estudio de Prevalencias de

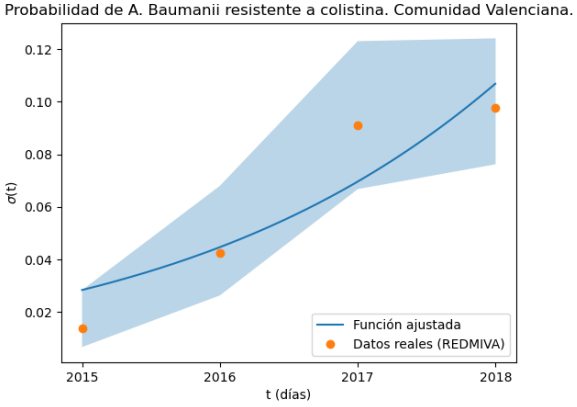


Fig. 5. Probabilidad $\sigma(t)$ de infección por cepa resistente dada una infección por *A. baumannii* resistente a colistina.

las Infecciones Nosocomiales de España (EPINE) [10]. En términos epidemiológicos, la prevalencia es la proporción de individuos de una población que presentan un determinado microorganismo o enfermedad infecciosa. Para este modelo se ha tomado el conjunto de prevalencias diarias $\rho = \{\rho_{IC}, \rho_{IN}, \rho_{IAB|IC}, \rho_{IAB|IN}\}$ de pacientes hospitalizados por infección de origen comunitario, de origen nosocomial, por infección causada por *A. baumannii* de origen comunitario y de origen nosocomial, respectivamente. También son de interés los tamaños muestrales de los infectados hospitalizados n_I, de los análisis microbiológicos realizados a pacientes con infección comunitaria n_{IC} y con infección nosocomial n_{IN}. Los valores de las prevalencias y los tamaños muestrales de los estudios EPINE se muestran en las Tablas III y IV. Como las prevalencias se aproximan a una distribución uniforme (test χ^2, p-valor $\simeq 1$), asumimos que las probabilidades de infección π son constantes en el tiempo.

TABLA III
PREVALENCIAS DE INFECCIONES INTRAHOSPITALARIAS. ESPAÑA, PERIODO 2015-2018 [10].

Año	ρ_{IC}	ρ_{IN}	$\rho_{IAB\|IC}$	$\rho_{IAB\|IN}$
2015	0.1828	0.0806	0.0042	0.0153
2016	0.1891	0.0792	0.0046	0.0155
2017	0.1853	0.0774	0.0046	0.0180
2018	0.1797	0.0715	0.0046	0.0114

TABLA IV
TAMAÑOS MUESTRALES DE INFECCIONES INTRAHOSPITALARIAS. ESPAÑA, PERIODO 2015-2018 [10].

Año	n_I	n_{IC}	n_{IN}
2015	15 052	5 512	4 302
2016	15 833	6 077	4 383
2017	16 201	6 124	4 512
2018	15 182	6 065	4 229

III. Calibrado del modelo

Con el calibrado se pretende estimar las probabilidades constantes π de infección intrahospitalaria desconocidas tales que den lugar en el entorno hospitalario a las prevalencias diarias $\hat{\rho}$ lo más aproximadas posibles a las prevalencias ρ conocidas. Al disponer de un modelo basado en agentes y controlar la dinámica de la población al completo, es posible identificar en cada instante temporal t el número de individuos con determinados estados y estimar fácilmente las prevalencias $\hat{\rho} = \left\{\hat{\rho}_{IC}, \hat{\rho}_{IN}, \hat{\rho}_{IAB|IC}, \hat{\rho}_{IAB|IN}\right\}$ como la fracción entre los casos de que presentan una determinada infección con respecto a los casos totales del estado de interés. Nótese que cualquiera de las prevalencias estimadas por el modelo $\hat{\rho}_x \in \hat{\rho}$ depende de las probabilidades π que se pretenden optimizar, así como del instante temporal t en el que se estiman.

Para calcular el error asociado a una simulación del modelo, definimos en primer lugar la función de distancia como

$$\begin{aligned} d\left(\hat{\rho}_x, \rho_x\right) &= d\left(\hat{\rho}_x, \left[\rho_x^i, \rho_x^s\right]\right) \\ &= \begin{cases} 0 & \hat{\rho}_x \in \left[\rho_x^i, \rho_x^s\right], \\ \min\left\{\left|\rho_x^i - \hat{\rho}_x\right|, \left|\rho_x^s - \hat{\rho}_x\right|\right\} & \hat{\rho}_x \notin \left[\rho_x^i, \rho_x^s\right]. \end{cases} \end{aligned} \tag{3}$$

donde $\hat{\rho}_x \in \hat{\rho}$ es la prevalencia estimada por el modelo y $\left[\rho_x^i, \rho_x^s\right]$ es el intervalo de confianza binomial al 95% [11] asociado a la prevalencia ρ_x conocida.

Una vez definida la distancia, definimos la función de error o *fitness* como

$$\varepsilon(\hat{\rho}, \rho) = \sum_{(\rho_x, \hat{\rho}_x) \in (\rho, \hat{\rho})} \frac{1}{T} \sum_{t=1}^{T} d\left(\hat{\rho}_x, \rho_x\right), \tag{4}$$

donde $T = \frac{365 \text{ días}}{1 \text{ año}} \times 4 \text{ años} = 1460$ días, el número de instantes temporales que se simulan en el periodo temporal de calibrado.

Formalmente, el problema de optimización que se resuelve en el calibrado se define del siguiente modo:

$$\pi^* = \arg\min_{\pi} \left\{\varepsilon\left(\hat{\rho}, \rho\right) : \hat{\rho} = \hat{\rho}(t; \pi), \rho = \rho(t)\right\} \tag{5}$$

Para ello, el proceso se ha llevado a cabo mediante dos subprocesos o módulos claramente diferenciados pero complementarios: el *simulador* y el *optimizador*. El simulador genera una población sintética aleatoria y simula el modelo basado en agentes con un conjunto de parámetros π dados por el optimizador, y calcula las prevalencias $\hat{\rho}$ estimadas en cada instante t y la función de error $\varepsilon(\hat{\rho}, \rho)$. El optimizador recibe los errores del simulador y genera nuevos valores de parámetros π en base a una versión del algoritmo de optimización Particle Swarm Optimization (PSO) [12]. La versión empleada del algoritmo PSO se ha programado con las siguientes especificaciones: *(i)* una probabilidad de desechar una partícula actualizada y sustituirla por otra obtenida aleatoriamente del 10% *(ii)* una probabilidad de mutación (intercambio de parámetros) del 10% *(iii)* si uno de los parámetros se encuentra muy cerca de los extremos del intervalo de búsqueda, se desecha y se reemplaza por un valor aleatorio dentro del intervalo. La implementación del simulador se ha realizado en el lenguaje de programación compilado Julia [13] de forma distribuida, y el optimizador se ha programado en lenguaje interpretado Python [14].

Se han llevado a cabo dos procesos completos de calibración, fijando un número máximo de 3000 simulaciones del modelo con diferentes parámetros generados por el algoritmo PSO. En el primer proceso se ha explorado el intervalo $[0, 1]$ para todas las probabilidades π, con la finalidad de localizar las zonas del espacio de parámetros con menor error. En el segundo, se han acotado los intervalos de búsqueda de parámetros en base a los resultados del primer proceso. Para garantizar la reproducibilidad de los resultados sin renunciar a la aleatoriedad del modelo, se ha incorporado en cada simulación una semilla para la generación de valores aleatorios en el modelo.

IV. Resultados y discusión

Tras el proceso de calibrado del modelo, se han obtenido los parámetros óptimos de la Tabla V, que arrojan el mejor error del calibrado. Observamos que para mantener la prevalencia de infecciones de origen comunitario y nosocomial en el entorno hospitalario, la probabilidad diaria de nuevo ingreso por infección comunitaria es de aproximadamente el 18.37%, mientras que la probabilidad diaria de infección nosocomial es del 2.57%. Esta gran diferencia se explica debido a que un paciente tan solo puede presentar infección comunitaria en el momento del ingreso, mientras que la infección nosocomial la puede contraer cada día, durante toda la estancia. Por lo tanto, la probabilidad de sufrir una infección nosocomial del paciente se "distribuye" en los instantes temporales de su estancia, y es más baja. Por otro lado, observamos que la probabilidad de infección por *A. baumannii* nosocomial es ligeramente más alta que probabilidad de infección por *A. baumannii* comunitaria, lo que se corresponde con los datos de prevalencia y la evidencia de la preferencia de esta bacteria por el entorno hospitalario. En estos dos últimos casos, la probabilidad de infección diaria es muy baja, del orden de 10^{-3}, lo que demuestra la capacidad del modelo para capturar la evolución de la bacteria a pesar de que las infecciones son eventos aleatorios muy poco probables.

TABLA V
Parámetros óptimos π^* del modelo basado en agentes.

Parámetro	Valor
$\pi^*_{IC\|CH}$	0.183745
$\pi^*_{IN\|H}$	0.025784
$\pi^*_{IAB\|IC}$	0.003065
$\pi^*_{IAB\|IN}$	0.009559
Error $\varepsilon(\hat{\rho}, \rho)$	0.010502

La Fig. 6 muestra el resultado del calibrado. En todas las curvas se observa el efecto de la aleatoriedad del modelo, y cómo el algoritmo de optimización ha logrado estimar

las probabilidades π^* del modelo para conseguir ajustar las prevalencias $\hat{\rho}_x \in \hat{\rho}$ a los intervalos de confianza anuales $\left[\rho_x^i, \rho_x^s\right]$. En todas las prevalencias calibradas se observa cómo el modelo ha capturado la probabilidad que mejor se ajusta a todos los tramos del intervalo de confianza.

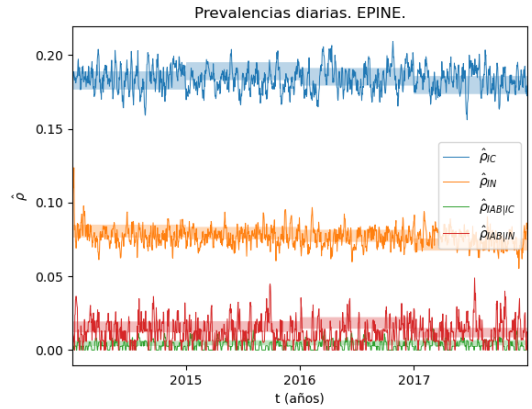


Fig. 6. Prevalencias diarias estimadas $\hat{\rho}_x \in \hat{\rho}$ frente a los intervalos de confianza $\left[\rho_x^i, \rho_x^s\right]$.

El proceso de búsqueda de los parámetros óptimos y la minimización del error del algoritmo PSO se muestra en la Fig. 7. En la función observamos como el algoritmo alcanza el mejor error sobre la iteración $k = 1700$.

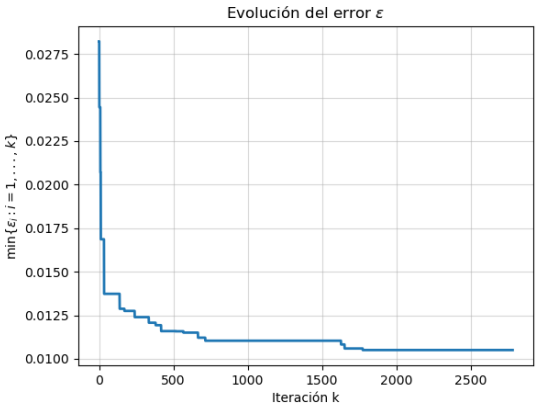


Fig. 7. Evolución del error en el algoritmo PSO.

V. Conclusiones

En este trabajo se ha propuesto un modelo basado en agentes de la evolución de la resistencia a la colistina en la bacteria *A. baumannii*. El modelo se ha aplicado sobre una población sintética de individuos generada a partir de los datos demográficos de la ciudad de Valencia, cuyos estados evolucionan aleatoriamente en función de un conjunto de probabilidades.

Para estimar los parámetros desconocidos, se ha llevado a cabo un proceso de calibrado, tomando como referencia datos epidemiológicos de prevalencias. Al tratarse de un modelo basado en agentes, se ha podido acceder a la información de todos los individuos en cada paso temporal y calcular las prevalencias a partir del conteo de individuos con determinados estados. Se ha definido una función de *fitness* que calcula el error en función de la aproximación de las prevalencias estimadas por el modelo con los intervalos de confianza de las prevalencias reales. La optimización de la función de *fitness* se ha realizado mediante una versión del algoritmo de optimización PSO.

De los resultados se concluye que prevalencias estimadas por el modelo tras el proceso de calibrado se ajustan correctamente a los intervalos de confianza de las prevalencias reales. Por lo tanto, conocidos todos los parámetros del modelo, y como línea futura, es posible realizar un estudio completo de la evolución de la bacteria *A. baumannii* a futuro, deduciendo en cada periodo temporal los casos resistentes, las características demográficas de los individuos afectados, los costes sanitarios, etc. Con ello, se espera disponer de una previsión de la evolución de la AMR en *A. baumannii* que apoye a la toma de decisiones clave ante este problema de salud pública.

Referencias

[1] W. H. Organization. (2020, Oct.) Antimicrobial resistance. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/antimicrobial-resistance

[2] R. Wise, "Antimicrobial resistance: priorities for action," *Journal of Antimicrobial Chemotherapy*, vol. 49, no. 4, pp. 585–586, 04 2002. [Online]. Available: https://doi.org/10.1093/jac/49.4.585

[3] F. D. Lowy, "Antimicrobial resistance: the example of staphylococcus aureus," *Journal of Clinical Investigation*, vol. 111, no. 9, pp. 1265–1273, May 2003. [Online]. Available: https://doi.org/10.1172/jci18535

[4] J. O'Neill. (2016, May) Tackling drug-resistant infections globally: final report and recommendations. [Online]. Available: https://amr-review.org/sites/default/files/160518_Final paper_with cover.pdf

[5] G. Valenciana. Portal estadístico. estimaciones de población. [Online]. Available: https://pegv.gva.es/es/estimaciones-de-poblacion-de-la-comunitat-valenciana

[6] M. de Sanidad. Portal estadístico. área de inteligencia de gestión. [Online]. Available: https://pestadistico.inteligenciadegestion.mscbs.es/publicoSNS/N/siae

[7] R. H. Sunenshine, M.-O. Wright, L. L. Maragakis, A. D. Harris, X. Song, J. Hebden, S. E. Cosgrove, A. Anderson, J. Carnell, D. B. Jernigan *et al.*, "Multidrug-resistant acinetobacter infection mortality rate and length of hospitalization," *Emerging infectious diseases*, vol. 13, no. 1, p. 97, 2007.

[8] G. Valenciana. Red de vigilancia microbiológica de la comunitat valenciana. [Online]. Available: http://www.sp.san.gva.es/sscc/

[9] N. Fallach, Y. Dickstein, E. Silberschein, J. Turnidge, E. Temkin, J. Almagor, Y. Carmeli, D.-A. Consortium *et al.*, "Utilising sigmoid models to predict the spread of antimicrobial resistance at the country level," *Eurosurveillance*, vol. 25, no. 23, p. 1900387, 2020.

[10] I. de Investigación Sanitaria Puerta de Hierro Segovia de Arana. Estudio de prevalencia de las enfermedades nosocomiales en españa (epine). [Online]. Available: https://epine.es/resultados/espania

[11] L. D. Brown, T. T. Cai, and A. DasGupta, "Interval estimation for a binomial proportion," *Statistical science*, pp. 101–117, 2001.

[12] F. Marini and B. Walczak, "Particle swarm optimization (PSO). a tutorial," *Chemometrics and Intelligent Laboratory Systems*, vol. 149, pp. 153–165, Dec. 2015. [Online]. Available: https://doi.org/10.1016/j.chemolab.2015.08.020

[13] The julia programming language. [Online]. Available: https://julialang.org/

[14] Pyhton. [Online]. Available: https://www.python.org/

A Blood Glucose classification and prediction system by Neural Networks

Alvaro Delgado Gutierrez
Facultad de Informática
Universidad Complutense de Madrid
Spain
absys@ucm.es

Alejandro Varela Lorenzo
Facultad de Informática
Universidad Complutense de Madrid
Spain
absys@ucm.es

Félix Tena
Facultad de Informática
Universidad Complutense de Madrid
Spain
absys@ucm.es

José Manuel Velasco
Universidad Complutense de Madrid
Madrid, España
mvelascc@fis.ucm.es

J. Ignacio Hidalgo
Universidad Complutense de Madrid
Madrid, España
hidalgo@dacya.ucm.es

***Abstract*—This paper compares the performance of two different neural networks, trained from different approaches of data, for blood-glucose prognosis on 30, 60, 90 and 120 minutes prediction horizon. We compare their performance using the most common metrics in blood glucose prediction and rank the best performing ones using three methods devised for the statistical comparison of the performance of multiple algorithms: scmamp, model confidence set, and superior predictive ability. Our analysis highlights those models with the highest probability of being the best predictors, estimates the increase in error of the models that perform more poorly with respect to the best ones, and provides a guide for their use in clinical practice.**

***Index Terms*—Diabetes, Glucose Prediction, Neural netwoks**

I. Introduction

Diabetes is a chronic disease that is irreversible and affects the normal transformation of sugar in energy. Insulin, a hormone made by the pancreas, helps glucose from food get into the cells to be used for energy. Sometimes, the body does not produce enough insulin or does not use insulin correctly, so glucose stays in the bloodstream and doesn't reach the cells. Over time, maintaining high levels of glucose in the blood can cause several health problems So, a goof management of the disease is necessary to avoid acute complications. We can differentiate two main types of diabetes:

- **T1DM (type 1):** This type is the less common within the patients of diabetes, a tenth of the total. It is an autoimmune illness that attacks the pancreatic cells which generate insulin, making them dysfunctional. Thus, the patients need to supply themselves with daily doses of insulin that replace those the metabolic system should generate.
- **T2DM (type 2):** With diabetes type 2, the metabolic system is able to generate insulin, but the immune system creates resistance to it, resulting in a similar outcome as in T1DM. Anyways, people with T2DM do not always need to supply themselves with daily doses of insulin. Normally, it is enough if they have a healthy routine and diet. This type of diabetes appears along the lifetime, and can be prevented or delayed with a healthy lifestyle (exercise, healthy diets...).

To maintain normal levels of glucose, the patients need to have control of those levels every time, taking into account the physical effort, diet and other factors such as alcohol intakes, level of stress, etc. The first issue is to know the level of blood glucose (BG), or at least a good approximation to the real value. To measure the levels of glucose it is usual to have a manual glucose meter (GM) or a continuous glucose monitoring system (CGMS). This is a tedious task that is done on a daily basis by the patient and that provokes an extra stress. Patients need to know the glucose level to decide the ammount of insulin to be injected or if they need to take a corrective action, such as injecting insulin or eating some food. To help with this routine, new methods are being investigated. One of them is the prediction of glucose levels, which could potentially allow patients to anticipate when there is a risk of hypoglycemia or hyperglycemia.

However, glucose levels prediction is a extremely difficult task because of the great amount of factors that are involved. Furthermore, predictions must be really accurate, since a wrong dose of insulin, taken from a wrong prediction, can cause health problems on the patient.

Nowadays, using new technologies such as Artificial Intelligence (AI) or Machine Learning (ML), these calculations can be supported and eased by the application of prediction systems. The field of AI and ML is very large, providing scientists with several different tools to create forecasting algorithms. In this paper, we focus on the creation and use of Neural Networks (NN) for glucose level prediction. To create this systems, we explored different types of Neural Networks (NN), ranging from regular numeric NN to graphic NN applications, which are vastly known in the world of data scientists.

However, these algorithms are always a difficult and hidden process for the real users, diabetes patients, so in order to make a higher impact on people affected by this disease, we

will create an online application that will share all the power of NNs with the final patients, using a clear and intuitive user interface[1]. In the meantime, patients will collaborate on the NN training process, as long as data provided by those using the application will allow us to develop a more heavily trained NN, thus improving its effectiveness in glucose levels forecasting. Referring to the numerical NN and according to the results, we can state that their performance for 30 and 60 minutes prediction is quite accurate and, for 90 and 120 minutes prediction, it throws promising results.

The rest of the paper is organized as follows. Section II summarize the works on the field of glucose prediction and NNs. Section III describes the particularities of the models. In Section IV we give details of the experimental results comparing two kinds of NN between them and using different data for training. Finally, Section V concludes the paper and exposes the future work.

II. Previous Work

This field of study is not new. Previously, numerous techniques have been tried for glucose forecasting. The greatest aim of this techniques is to achieve a permanent solution for diabetes, such as the development of an ”artificial pancreas”.

Through an artificial pancreas with an advanced prediction system along with measurement instrument such as a continuous glucose monitor, it could be possible to automate the insulin injection process in the same way as the pancreas of a healthy person works. Some of the techniques used for this prediction models are: Genetic Programming, Random Forest Regression, K-Nearest Neighbors or Grammatical Evolution [1] and, of course, Neural Networks [2]. There are several kind of Neural Networks and, due to the availability of new high performance computing architectures, the development and the applicability of them have increased exponentially during the last five years. The main types of NN are: Long Short Term memory (LSTM), Recurrent Neural Network (RNN), Multilayer Perceptron (MLP) and Convolutional Neural Network (CNN).

LSTM networks have achieved very good performance in modeling several time-dependent phenomena. This is why most of the approximations found in the literature for the prediction of Blood glucose Levels are LSTM, since BG Data is usually obtained as a time series from CGMs. Although most of the NN in the literature they are mostly devoted for time series predictions, there are also other Neural Network implementations following different schemes. In this section we made a revision of the recent solutions proposed for blood and interstitial glucose prediction using Neural Networks.

Allam et al (2011) [3] proposed to use CGM signals to train an RNN to predict future glucose concentration values, considering several prediction horizons. As in Tresp, the data used prevent the generalization of the model, to deal with data not used in training.

[1]A beta version of the web system can be accessed through glucnet.ucm.es

[4] develops a predictive model of BG to define future insulin therapy. The proposed architecture is called Deep Glucose Forecasting (DGF) and consists of two stacked LSTMs working in parallel. The first uses observed data and the second uses estimated data. The output of both branches is concatenated in a full connected layer. It performs justified data pre-processing because insulin and carbohydrate doses are 100 times higher than glucose measurements. In order to ensure that all values contribute equally to training these signals are re-scaled using their maximum and minimum values. It uses datasets obtained from UVA/PADOVA and real patients (PADOVA clinical center collected during the *pa@home* Project experiments). Two different scenarios are used to simulate realistic intra-subject changes. To evaluate the results it uses the metrics Coefficient of Determination, Fittint Index, Root mean square error. The proposed solution is obtained as an average of the results obtained with 100 virtual adults. To improve the model, the weights of the LSTM are adjusted for a specific patient. The main limitation of the proposal is that the network is trained with measurements with noise, but the output has to be a signal without noise. Since the noise from the CGM measurements cannot be neglected, the network should try to reduce the noise from the output to improve the prediction. This is achieved by applying exponential filtering. The proposed architecture obtains a performance similar to the current state of the art, both in in-silico patients, and in real patients, considering several prediction horizons 5, 10,...60.

Another interesting work is explained in [5]. This paper predicts glucose levels with a horizon of up to one hour. It uses a LSTM that only uses CGM values. The output of the model is the prediction plus the μ and σ^2 of the series, i.e. it indicates the certainty of the prediction which helps the users to interpret the result. It does not pre-process data except to multiply the glucose values by 0.01 to indicate that these values are in a range that can be predicted, therefore, when CGM values fail (there are gaps) it does not extrapolate values. He explains the method he has followed to select the hyperparameters of the LSTM network. It uses as dataset the Ohio T1DM Dataset for Blood Glucose Level Prediction of which it uses only two patients (Marling and Bunescu, 2018). Evaluates the prediction using the RMSE and Clarke error grid with irregular results.

Mirshekarian et al. [6] proposes an LSTM architecture for predicting blood glucose levels at 30 and 60 minutes. Perhaps the most interesting part of the paper is how they decide the architecture of the LSTM: they start from a physiological model and look for the dependencies between different parameters of the physiological model. Then, they look for an LSTM architecture that maintains the dependencies of the physiological model. Their proposal indicates that the obtained architecture can be easily extended to other parameters. The architecture contains three layers, the input one, the hidden one, which is an LSTM, and the full connected output one. This last one has two additional nodes, one for the prediction at 30 minutes and the other at 60 minutes. They have assembled a database containing approximately 1600 days of actual patient

information. This data includes insulin, food, exercise and sleep. They perform a pre-processing: the BG is scaled by 0.01, the rest of the values are normalized to [0.1]. Lost BG values are interpolated linearly, however, they are not used as predictive targets during training. Compare the results with those obtained with the Support Vector Regression (SVR) indicating that they are competitive. They also claim that the best thing about the model is that it can incorporate new physiological parameters without the need for sophisticated manual engineering.

The objective of the work in [7] is the application of machine learning to the prediction of glycemia levels that can be used to generate alerts. In the article the authors try to demonstrate that the metric should be particularized for the range in which the highest accuracy is desired. Usually the accuracy is computed over the total glycemic range which can lead to erroneous models. One solution to this problem is to break the glycemic range into significant parts and analyze each subrange independently. The article also tries to demonstrate that a pre-processing of the data is necessary to take into account the imbalances that appear in the data, since most of the time the patients are in a region where the glycemia is normal. An adaptation of standard pre-processing techniques has been made for imbalanced labelled data. The latter increased accuracy in particular glycemic range subranges. In the experiments it uses as dataset the last six patients incorporated in the Ohio T1D dataset. In total, the work studies the behavior of 10 regression models among which the Multilayer Perceptron and four oversampling techniques to deal with data imbalances stand out: Random Oversampling, Synthetic Minority Oversampling Technique, Adaptive Synthetic Oversampling Technique. It uses two metrics to evaluate the performance of Mean Absolute Relative Difference (MARD) prediction and Clarke Error Grid Analysis. The MARD specifies it for four different cases: all regions, regions with hypoglycemia, regions with normoglycemia, and with hyperglycemia. To carry out the study, the 10 machine learning techniques proposed are combined with the 4 oversampling options, giving rise to 40 different combinations. The results show that if a hypoglycemic alert is desired, a trained MLP in conjunction with an oversampling method produces the highest accuracy. Therefore, it can be deduced that the choice of the best algorithm depends on the glycemic range.

Convolutional NNs have also been explored to predict the BG level. Although more used in image processing CNN can be adapted for this problem. For instance (Li et al., 2018) [8] transformed time series of glucose level, carbohydrate and insulin, by concatenating them and preprocessing using deep convolutional networks: Then the recurrent LSTM layers accepts these features instead of the CGM measurements directly. The model is trained on 10 in silico adults patients from the UVA/Padova simulator.

Other works:

- Prediction of Blood Glucose Levels And Nocturnal Hypoglycemia Using Physiological Models and Artificial Neural Networks [9]
- Predicting hypoglycemia in diabetic patients using time-sensitive artificial neural networks [10]
- Hypoglycemia prediction using extreme learning machine (ELM) and regularized ELM [11]
- Evolvable rough-block-based neural network and its biomedical application to hypoglycemia detection system [12]
- Artificial neural network algorithm for online glucose prediction from continuous glucose monitoring [2]

III. Methodology

In this project we are faced with a multivariate regression problem. The objective is to predict the blood glucose level that a patient with diabetes will have in the near future based on several input variables. The input data set can be organized as a multivariate time series to feed a LSTM network or as a classic supervised data set to train a Feed Forward network.

Because some patients have fewer input variables than others, we have worked with different models depending on the available input parameters. Thanks to this, we can evaluate the training performance as a function of the training data set and draw some conclusions about it.

Table I shows a summary of the 4 types of data sets and the input variables that correspond to each one, whereas Table II explains the different input variables that we had available.

TABLE I
Datasets

Model	Patients	Parameters
Insulin and Basic Values (IBV)	3	Glucose, Insuline, Steps, Heart rate Calories, Basal rate
Only Basic Values (OBV)	10	Glucose, Steps, Heart rate, Calories
Glucose and Basal Values (GBV)	9	Glucose, Basal Rate
Only Glucose (OG)	16	Glucose

Thus, we have four different types of data sets depending on the input variables. In addition to this combination, we have two more levels:

- If we use Transfer Learning or just a custom model, subsection III-C.
- The two types of neural networks that we have used: the Feed Forward Network (subsection III-A) and the LSTM (subsection III-B).

So, in total, we have trained 16 different models.

Before presenting and discussing the experimental results in section IV, we are going to describe the metrics that we have used in order to measure the efficiency and behaviour of our model: the Clarke Error Grid III-D and the RMSE III-E.

A. Feed Forward Network

Firstly the study collects results from a basic neural network, specifically a feed forward neural net implemented with Keras. This initial neural network is configured with the input layer (data supervised), one hidden dense layer with a hyperbolic tangent activation function that expects rows of data with the number of columns of the supervised data-set, and one output dense layer with one node.

- 190 epochs
- Tanh activation function
- Adam optimizer
- Input shape: (1, number of columns of the data-set)
- Mean absolute error as loss function

B. LSTM Network

FF neural networks are not specially designed to deal with time series data and therefore will not succeed in finding states and patterns that may occur over time. Like RNNs, LSTMs have feedback connections which will help them process sequences of data, but the latter, unlike the former, can keep their state hidden over a long time and find patterns at different time intervals.

Each component of an LSTM network has three gates that regulate the flow of information in and out of the cell, which is the processing unit.

C. Transfer Learning

We have opted for two different types of training. In the first case, the models for each patient are trained only with the patient's data. This is a personalized or custom model.

In the second option, the network of a patient is first trained with the data of the rest of patients and finally with the patient's data. This way, the knowledge acquired on the set of patients is attempted to be transferred to the model of a particular patient. This is an interesting option that can help to mitigate the problem of the scarcity of data and lead to more robust models.

D. Clarke Error Grid

Clarke error grid: This Clarke graphic visualizes black points that represent the real test target (axis x) and the predicted values (axis y). In a first instance, the closer the points are allocated around the diagonal (which means that predictions are equals to real test set), the more precise the model will be. This Clarke graphic displays different zones: A, B, C, D, E. These zones represent the validity of the model prediction. Our goal is that most of the predictions fall in the zones A and B, and as few points as possible in the zones C, D, and E. A prediction that falls in zone C produces an unnecessary treatment. Predictions that fall in zones D and E lead to erroneous treatments that can produce serious complications for the patient. An example of an empty Clarke Error Grid can be seen in Figure 1.

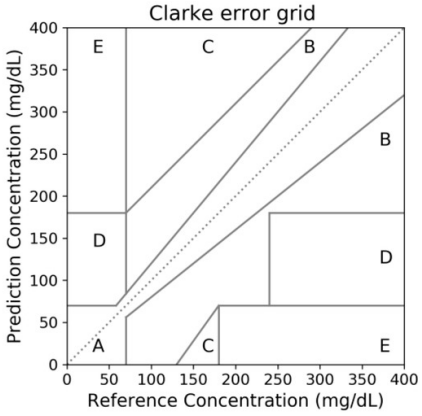


Fig. 1. Clarke Error Grid

E. Root Mean Square Error

RMSE. To have more measurements apart from the Clarke image and the number of points in each zone, we calculate the RMSE i.e root mean square error, following the formula in Equation 1. Due to the RMSE is calculated over glucose values that usually range between 40 and 500 ml/dl, an RMSE around 30 mg/dl is an indicator of acceptable predictive performance.

$$RMSE = \sqrt{\frac{1}{n}\Sigma_{i=1}^{n}\left(\frac{d_i - f_i}{\sigma_i}\right)^2} \quad (1)$$

IV. Results

In Tables III and IV we can see the different results for the FF and the LSTM networks respectively. On the left, we have the results for the test phase, whereas on the right, the results for the training phase are shown. On the rows, we distinguish among the different data sets and the different forecasting time horizons.

In view of these results, we can draw the following conclusions distinguishing among the different prediction horizons:

- For a Forecasting Horizon of 30 minutes:
  - The worst results in terms of RMSE correspond to the IBV data set (data set with the highest number of input variables), having the other data sets very similar values and being OBV (only basic variables) the best performing for both FF and LSTM networks.
  - However, we find the opposite situation when we look at the number of predictions that are placed in the dangerous C, D and E zones. Although the results are very similar, in this case, IBV and OBV have a slight advantage over the others, for FF and LSTM respectively.
  - In the case of the number of points falling in zones A and B, the results are very good even for the four data sets, all of them being between 98% and 99%.
- For a Forecasting Horizon of 60 minutes:
  - The worst results in terms of RMSE correspond again to the IBV data set. The other data sets have

TABLE II
MEANING OF EACH PARAMETER FROM ALCALÁ DE HENARES PATIENTS DATASETS

Column	Meaning	Measuring Device	Note
Basal Rate (U/h)	Basal rate	Medtronic Pump	Every 5 min if it is ok
Bolus Type	Bolus type	Medtronic Pump	anytime,
Bolus Volume Delivered (U)	Bolus Units	Medtronic Pump	anytime,
Sensor Glucose (mg/dL)	The sensor glucose value	CGM Medtronic	Every 5 min if it is ok
Hora	Time	Anyone	
Calories	Calories from Fitbit	Fitbit	anytime,
Heart Rate	HR from Fitbit	Fitbit	anytime,
Status	Sleeping state from Fitbit	Fitbit	anytime,
Steps	Steps from fitbit	Fitbit	anytime,
Glucose history (mg/dL)	Automatic glucose level	Freestyle	Every 15 min if it is ok
Glucose reading (mg/dL)	Manual glucose level from FreeStyleLibre	Freestyle	anytime,
Rapid-acting insulin without numerical value	Bolus Mark	Freestyle	anytime,
Rapid acting insulin (units)	Bolus units	Freestyle	anytime,
Food with no numerical value	Food intake	Freestyle	anytime,
Slow-acting insulin without numerical value	Slow action insulin timestamp	Freestyle	anytime,
Slow-acting insulin (units)	Units of slow action insulin	Freestyle	anytime,
Notes	Notes	Freestyle	anytime,
Glucose (mg/dL)	Glucose Meter from FreeStyle Libre	Freestyle	anytime,
Previous time	Time change	Freestyle	anytime,
Time updated	Time change	Freestyle	anytime,
Sensor Calibration BG (mg/dL)	Glucose calibration	Extern glucometer	anytime,
BG Reading (mg/dL)	Glucose calibration	Extern glucometer	anytime

very similar values and being OBV (only basic variables) the best performing.
  - Here, for zones C, D and E and in the case of the custom model, the best OBV data set. While for Tranfer Learning and/or LSTM, it is GBV.
  - For zones A and B, again we find that there is no option that has a significant advantage, having lowered the values to a range between 84% and 94%.
- For a Forecasting Horizon of 90 minutes:
  - For this prediction horizon we found that for both the RMSE and the Clarke Error Grid zones, the best results are obtained by the data sets with the lowest number of variables: GBV and OG. This is applicable for both network models (FF and LSTM) as well as for both types of training (custom and transfer learning).
  - We could say that in the case of FF, GBV has a slight advantage over OG. While for LSTM, the opposite is true.
  - The percentage of points located in zones A and B remains between 80% and 92%.
- Finally, for a Forecasting Horizon of 120 minutes:
  - For the farthest prediction horizon, the best results are clearly for the simplest data set, OG. This is true for both FF and LSTM, using a custom model as a transfer learning model.
  - In this case, the maximum percentage of points in the safe zones A and B always falls below 90%.

V. CONCLUSIONS AND FUTURE WORK

Summarizing the results analyzed in Section IV, we find several ideas that can guide future research.

- There does not seem to be a problem of overfitting, since the results do not differ significantly in the test phase with respect to the training phase, especially with respect to RMSE. This may be due to the fact that the length of our data sets is not sufficiently large.
- In general, we can say that the shorter the prediction horizon, the better the data sets with a larger number of input variables perform. While for longer prediction horizons, the advantage is for data sets with a smaller number of input variables, for both FF and LSTM.
- Our intuition prior to the experimental phase told us that the more input variables we used, the better the results would be. However, the data point in the opposite direction. This counter-intuitive fact may again be, again, due to the length of our data sets. The small number of available samples is insufficient to train the neural networks with such a larger number of input variables and, therefore, a larger number of neurons in the first layer.

In the future, our intention is to apply different forecasting and ensembles techniques such as KNN, random forest, gradient boosting... and compute more metrics in order to develop a better understanding of the results.

ACKNOWLEDGMENT

Trabajo financiado por Comunidad de Madrid y de los fondos FEDER a través del proyecto S2017/BMD3773 (GenObIA-CM) y del proyecto de Doctorado Industrial IND2020/TIC-17435, y por el Ministerio de Economía y Competitividad bajo el proyecto RTI2018-095180-B-I00.

REFERENCES

[1] José Ignacio Hidalgo, J. Manuel Colmenar, Gabriel Kronberger, Stephan M. Winkler, Oscar Garnica, and Juan Lanchares. Data based prediction of blood glucose concentrations using evolutionary methods. *J. Medical Systems*, 41(9):142, 2017.

TABLE III
FEED FORWARD (FF) NEURAL NETWORK

		Custom Model						Transfer Learning					
Dataset	**Horizon**	**RMSE**	**A**	**B**	**C**	**D**	**E**	**RMSE**	**A**	**B**	**C**	**D**	**E**
IBV	30	30.15	82.67	16.67	0.00	0.67	0.00	27.14	86.00	12.67	0.00	1.33	0.00
	60	52.58	52.33	41.00	0.67	5.67	0.33	51.05	56.33	38.00	0.33	5.33	0.00
	90	68.53	38.00	50.67	1.67	9.33	0.33	68.18	37.33	50.33	0.67	10.67	0.00
	120	77.27	34.33	49.67	1.00	13.67	1.00	80.19	28.00	51.33	1.33	19.33	0.67
OBV	30	25.03	84.20	14.60	0.00	1.10	0.00	19.18	88.10	9.60	0.00	2.10	0.00
	60	41.24	62.60	31.80	0.40	4.70	0.20	38.68	60.90	30.70	0.50	7.60	0.10
	90	52.35	50.30	40.70	0.80	7.70	0.30	51.37	49.80	39.40	1.50	9.00	0.30
	120	59.48	43.80	44.30	0.90	10.50	0.50	60.09	43.30	43.80	2.30	10.20	0.50
GBV	30	27.47	82.56	16.22	0.00	1.11	0.00	23.91	87.67	11.67	0.00	0.78	0.00
	60	43.10	62.44	33.44	0.11	3.56	0.11	41.98	62.11	33.44	0.22	4.00	0.00
	90	52.83	52.44	40.44	0.56	6.56	0.22	56.36	46.44	44.56	0.44	7.89	0.56
	120	60.34	44.44	46.33	0.67	7.89	0.33	74.40	30.33	53.22	1.67	11.67	2.89
Glucose	30	26.33	82.31	16.38	0.06	1.19	0.00	20.12	88.63	9.94	0.06	1.25	0.00
	60	41.35	62.63	32.31	0.31	4.50	0.19	37.35	64.56	28.81	0.56	5.88	0.13
	90	50.92	52.06	39.94	0.56	7.00	0.25	49.09	53.50	37.44	1.38	7.25	0.44
	120	57.26	45.31	44.31	0.63	9.50	0.31	56.64	46.81	42.50	2.00	8.38	0.44

TABLE IV
LONG SHORT TERM MEMORY (LSTM) NEURAL NETWORK

		Custom Model						Transfer Learning					
Dataset	**Horizon**	**RMSE**	**A**	**B**	**C**	**D**	**E**	**RMSE**	**A**	**B**	**C**	**D**	**E**
IBV	30	41.43	77.33	19.33	2.00	1.33	0.00	27.56	84.67	14.00	0.00	2.00	0.00
	60	63.59	51.33	38.67	3.33	6.67	0.00	51.05	56.67	36.33	1.00	6.00	0.00
	90	72.27	38.00	48.67	1.33	11.33	0.33	70.31	39.67	48.00	1.67	9.67	1.00
	120	89.94	30.33	48.33	4.00	15.67	1.67	84.61	27.33	54.67	2.00	15.00	0.67
OBV	30	25.51	83.70	15.20	0.00	1.00	0.00	20.14	86.3	11.2	1	1.5	0.00
	60	42.33	61.70	32.70	0.60	4.80	0.20	40.56	62.90	33.30	0.80	4.60	0.10
	90	52.48	50.40	40.60	0.70	7.90	0.30	53.60	48.90	42.40	1.50	6.00	0.50
	120	59.45	43.20	44.50	0.90	10.50	0.50	58.59	40.30	43.80	5.30	9.1	1.50
GBV	30	27.40	82.89	15.89	0.00	1.11	0.00	21.4	84.5	12.2	0.50	1.8	1.00
	60	43.99	62.56	32.89	0.44	3.78	0.11	37.88	64.2	31.5	0.22	3.57	0.08
	90	53.75	51.78	40.89	0.56	6.11	0.22	57.36	46.44	44.56	0.44	7.89	0.56
	120	60.52	45.22	44.78	0.56	8.44	0.56	69.60	30.33	53.22	1.67	11.67	2.89
Glucose	30	26.96	81.69	17.00	0.06	1.12	0.00	18.92	84.34	13.89	0.06	2.25	0.00
	60	40.99	63.62	31.50	0.44	4.19	0.25	34.5	64.56	28.81	0.56	5.88	0.12
	90	50.67	51.62	40.50	0.50	6.88	0.31	51.59	53.50	37.44	1.38	6.14	0.86
	120	56.54	45.75	44.00	0.50	9.31	0.31	60.20	44.81	43.50	2.12	8.80	0.56

[2] Carmen Pérez-Gandía, A Facchinetti, G Sparacino, C Cobelli, EJ Gómez, M Rigla, Alberto de Leiva, and ME Hernando. Artificial neural network algorithm for online glucose prediction from continuous glucose monitoring. *Diabetes technology & therapeutics*, 12(1):81–88, 2010.

[3] Fayrouz Allam, Zaki Nossai, Hesham Gomma, Ibrahim Ibrahim, and Mona Abdelsalam. A recurrent neural network approach for predicting glucose concentration in type-1 diabetic patients. In *Engineering Applications of Neural Networks*, pages 254–259. Springer, 2011.

[4] Eleonora Maria Aiello, Giuseppe Lisanti, Lalo Magni, Mirto Musci, and Chiara Toffanin. Therapy-driven deep glucose forecasting. *Engineering Applications of Artificial Intelligence*, 87:103255, 2020.

[5] John Martinsson, Alexander Schliep, Björn Eliasson, Christian Meijner, Simon Persson, and Olof Mogren. Automatic blood glucose prediction with confidence using recurrent neural networks. In *3rd International Workshop on Knowledge Discovery in Healthcare Data, KDH@ IJCAI-ECAI 2018, 13 July 2018*, pages 64–68, 2018.

[6] Sadegh Mirshekarian, Razvan Bunescu, Cindy Marling, and Frank Schwartz. Using lstms to learn physiological models of blood glucose behavior. In *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 2887–

[8] Kezhi Li, John Daniels, Chengyuan Liu, Pau Herrero, and Pantelis Georgiou. Convolutional recurrent neural networks for glucose prediction. *IEEE journal of biomedical and health informatics*, 24(2):603–613, 2019.

2891. IEEE, 2017.

[7] Michael Mayo, Lynne Chepulis, and Ryan G Paul. Glycemic-aware metrics and oversampling techniques for predicting blood glucose levels using machine learning. *PloS one*, 14(12), 2019.

[9] Arthur Bertachi, Lyvia Biagi, Iván Contreras, Ningsu Luo, and Josep Vehí. Prediction of blood glucose levels and nocturnal hypoglycemia using physiological models and artificial neural networks. In *KHD@ IJCAI*, pages 85–90, 2018.

[10] Khouloud Safi Eljil, Ghassan Qadah, and Michel Pasquier. Predicting hypoglycemia in diabetic patients using time-sensitive artificial neural networks. *International Journal of Healthcare Information Systems and Informatics (IJHISI)*, 11(4):70–88, 2016.

[11] Xue Mo, Youqing Wang, and Xiangwei Wu. Hypoglycemia prediction using extreme learning machine (elm) and regularized elm. In *2013 25th Chinese Control and Decision Conference (CCDC)*, pages 4405–4409. IEEE, 2013.

[12] Phyo Phyo San, Sai Ho Ling, Hung Nguyen, et al. Evolvable rough-block-based neural network and its biomedical application to hypoglycemia detection system. *IEEE transactions on cybernetics*, 44(8):1338–1349, 2013.

XIV Congreso Español de Metaheurísticas, Algoritmos Evolutivos y Bioinspirados

Sesión especial:

Estrategias de implementación eficiente de heurísticas y metaheurística



Heurísticas para la mejora de la mantenibilidad de proyectos *software*

Javier Yuste
Universidad Rey Juan Carlos
Móstoles, España
javier.yuste@urjc.es

Eduardo G. Pardo
Universidad Rey Juan Carlos
Móstoles, España
eduardo.pardo@urjc.es

Abraham Duarte
Universidad Rey Juan Carlos
Móstoles, España
abraham.duarte@urjc.es

***Resumen*—Alrededor del 75 % de los costes de un proyecto *software* se deben a su mantenimiento. En este proceso, la mayor parte del esfuerzo se dedica a la comprensión del *software*. Para mejorar la mantenibilidad y reducir los costes de los proyectos es crucial organizar el *software* dividiéndolo en módulos independientes. La consecución del objetivo de alcanzar una buena división modular se enuncia como un problema de optimización, denominado *Software Module Clustering Problem* (SMCP). Para ello, la estructura que representa la organización del *software* se modela utilizando grafos, en los que los elementos del proyecto se representan mediante vértices y las dependencias entre elementos, mediante aristas. Los vértices del grafo se agrupan en conjuntos representando una solución. El objetivo del SMCP es dividir los elementos en grupos de tal forma que se maximice la modularidad del grafo. En este trabajo, se evalúan 4 heurísticas diferentes para la construcción de soluciones para el SMCP. Estas heurísticas son comparadas en términos de calidad de las soluciones y tiempo de ejecución sobre un conjunto de proyectos *software* reales. Las soluciones obtenidas muestran mejor modularidad que la organización original de los proyectos *software* evaluados. Además, esta mejora en la modularidad de los proyectos es mayor cuanto más grande es el proyecto. Por último, se presenta una implementación eficiente para la evaluación de la calidad de las soluciones.**

***Palabras clave*—Modularización de *software*, Mantenimiento de *software*, Ingeniería del *software* basada en búsqueda, Algoritmos de agrupamiento**

I. Introducción

El ciclo de vida de un proyecto *software* está compuesto por diferentes procesos, como la definición de requisitos, el diseño del sistema, el desarrollo del *software* o su mantenimiento [1]. Durante el proceso de mantenimiento, el *software* sufre modificaciones en el código con el objetivo de corregir fallos, mejorar el *software*, prevenir problemas potenciales o adaptar el producto a cambios en el entorno. Con frecuencia, los costes del mantenimiento de un proyecto *software* suponen alrededor de un 75 % de los costes totales del proyecto [2, 3]. Dentro del proceso de mantenimiento, la mayor parte del esfuerzo se dedica a la comprensión del *software* existente. Esta tarea, además, es más complicada cuanto mayor es el tamaño del código y supone un obstáculo inicial para los desarrolladores que se van incorporando al proyecto.

Esta investigación ha sido parcialmente financiada por el Ministerio de Ciencia, Innovación y Universidades (Ref. PGC2018-095322-B-C22) y por la Comunidad de Madrid y los Fondos Europeos para Desarrollo Regional (Ref. P2018/TCS-4566)

Un sistema *software* se compone de un conjunto de elementos (clases, interfaces, ficheros, etc.) que están relacionados entre sí. Dos elementos están relacionados si al menos uno de ellos utiliza funciones o datos del otro. Por ejemplo, dos clases o ficheros A y B están relacionados si A utiliza una función de B o accede a una de sus variables. En ese caso, se dirá que A depende de B. La modularización del código de un proyecto *software* es el proceso de división de un sistema *software* en módulos (agrupaciones de elementos) independientes. Un *software* modular destaca por tener un bajo nivel de interdependencia entre módulos (acoplamiento) y un alto nivel de relaciones entre elementos que pertenecen a un mismo módulo (cohesión). Esta división del código facilita la comprensión de cada módulo del proyecto de manera independiente [4], la identificación de los módulos responsables de una funcionalidad determinada [5] y la navegación entre partes del proyecto [6]. Por el contrario, una mala organización del sistema puede propiciar la aparición de fallos en el código [7].

El problema de modularización de un sistema *software*, conocido como *Software Module Clustering Problem* (SMCP), es un problema de optimización en el que el objetivo es encontrar la organización del sistema con la mayor modularidad posible (la solución óptima) de entre todas las organizaciones posibles (soluciones factibles) [8]. La calidad de la organización del *software* se mide por las dos métricas introducidas anteriormente: su acoplamiento y su cohesión. Normalmente, el SMCP se modela como un problema de particionamiento de grafos. En este modelo, las relaciones del sistema *software* de entrada se representan mediante un grafo dirigido, conocido como *Module Dependency Graph* (MDG). En este grafo, los elementos del sistema son representados mediante vértices, mientras que las relaciones entre elementos se representan mediante aristas. El problema, entonces, consiste en dividir los vértices en grupos con el objetivo de maximizar la modularidad del grafo. Este problema es NP-difícil [9, 10]. Es decir, el tiempo necesario para encontrar y evaluar todas las soluciones posibles aumenta rápidamente con el tamaño de la entrada, por lo que la utilización de algoritmos aproximados es más apropiada que la utilización de algoritmos exactos.

En este trabajo, se evalúan 4 heurísticas para la construcción de soluciones en el SMCP, comparando la calidad de las soluciones obtenidas y el tiempo de ejecución necesario para encontrar las soluciones. Finalmente, se muestra cómo mejora

la modularidad de los proyectos tras aplicar las heurísticas mostradas con respecto a la organización original. Las heurísticas evaluadas pueden ser utilizadas, posteriormente, en el diseño de metaheurísticas para el SMCP. Además, se describe un método eficiente para la evaluación de la calidad de las soluciones.

El resto del trabajo se estructura de la siguiente forma. En la Sección II, se resumen algunos trabajos previos del estado del arte. En la Sección III, se describen los algoritmos heurísticos propuestos en este trabajo. En la Sección IV, se describe una implementación eficiente de la función objetivo utilizada. En la Sección V, se presenta un procedimiento de preprocesado de las instancias para reducir su tamaño. En la Sección VI, se comparan los resultados obtenidos en la experimentación. Finalmente, se exponen las conclusiones en la Sección VII.

II. Estado del Arte

El objetivo del SMCP es aumentar la modularidad de un sistema *software* para mejorar su mantenibilidad. Para evaluar la modularidad de un sistema, se han propuesto distintas medidas en el estado del arte, como *Modularization Quality* (MQ) [8], *Maximizing Cluster Approach* (MCA) [11] o *Equal-Size Cluster Approach* (ECA) [11]. Además, existen métricas que se basan en la similitud entre la solución propuesta y una descomposición del sistema realizada por expertos, como *MoJo* [12]. En este trabajo, se utiliza MQ como la función objetivo, siguiendo el ejemplo de otros trabajos en el área [11, 12, 13].

Algunos autores han propuesto diversos algoritmos basados en metaheurísticas poblacionales para abordar el SMCP. En [11], los autores compararon los resultados de distintos algoritmos genéticos con dos representaciones distintas de las soluciones (*Group Number Encoding* y *Grouping Genetic Algorithms*). En [14] y [15], los autores propusieron dos algoritmos basados en *Harmony Search*. Otros trabajos en el área presentaron algoritmos basados en metaheurísticas de inteligencia de enjambre, como *Artificial Bee Colony* [16], *Ant Colony Optimization* [17] o *Particle Swarm Optimization* [13].

Otras aproximaciones incluyen algoritmos basados en metaheurísticas trayectoriales. En [18], los autores propusieron un algoritmo basado en *Iterated Local Search*, que fue comparado favorablemente con otros algoritmos basados en metaheurísticas poblacionales. Esta propuesta fue posteriormente utilizada como punto de comparación en [19], donde los autores propusieron la utilización de un algoritmo basado en *Large Neighborhood Search*.

III. Heurísticas

En esta sección, se presentan las heurísticas propuestas en este trabajo. En particular, se comparan 4 heurísticas diferentes para el SMCP.

III-A. C1

La primera de las heurísticas planteadas presenta una reconstrucción aleatoria del grafo. Cada vértice del grafo se asigna a un grupo aleatorio entre 0 y $n-1$, donde n es el número de vértices del grafo. Es decir, el número de grupos posibles es igual al número de vértices totales. Al finalizar, se eliminan los grupos vacíos.

III-B. C2

Esta heurística presenta una construcción iterativa que parte de un grafo vacío. La solución se va construyendo de manera parcial, añadiendo vértices en cada iteración del algoritmo. Al comenzar, se selecciona un vértice al azar que se añade a un grupo vacío del grafo. Después, en cada iteración, se selecciona un vértice al azar de entre los vértices restantes (aún no asignados a ningún grupo) y se prueba a asignarlo a:

- Cada uno de los grupos existentes.
- Un grupo vacío.
- Un grupo vacío, probando a mover cada uno de los vértices ya presentes en la solución a este grupo vacío junto con el vértice seleccionado.

Al finalizar la iteración, se realiza el movimiento con el que la solución parcial resultante tiene mayor calidad. El proceso finaliza cuando todos los vértices han sido asignados a un grupo.

III-C. C3

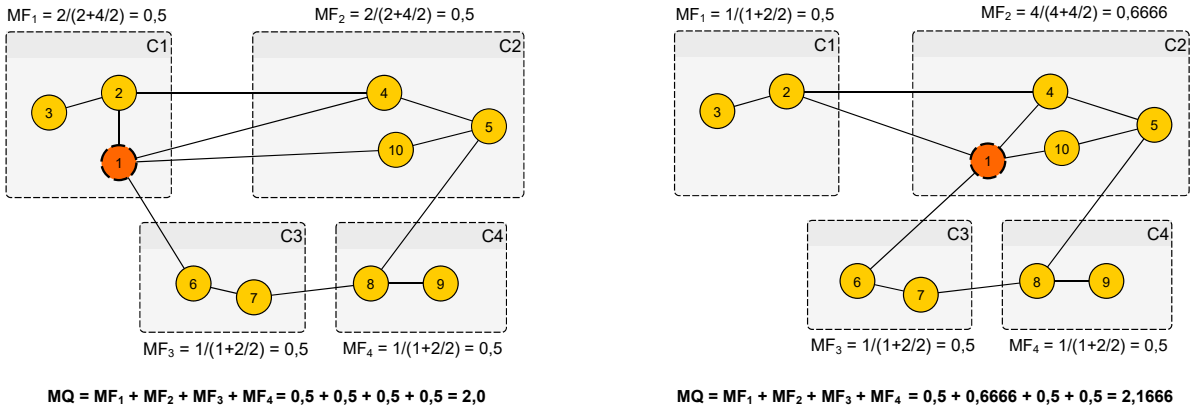
Este algoritmo, conocido como *Graph Growing Partitioning* [20], busca construir grupos alrededor de vértices seleccionados al azar. El proceso comienza seleccionando un vértice al azar, que es añadido a un grupo vacío. En cada iteración, se selecciona el vértice no asignado que tenga el mayor número de conexiones al grupo que se está construyendo. El vértice seleccionado se añade al grupo en construcción. Este proceso se repite hasta que el grupo en cuestión alcanza un tamaño predefinido. En ese momento, se reinicia el proceso para construir un nuevo grupo. El algoritmo finaliza cuando todos los vértices han sigo asignados a un grupo. El tamaño de los grupos es un parámetro que debe ser ajustado de manera experimental. En este caso, el número de grupos se ha limitado a un valor aleatorio de entre el 25 % y el 75 % del número de vértices de la instancia.

III-D. C4

En este algoritmo, se parte de una solución trivial en la que cada vértice se encuentra en un grupo diferente. Es decir, todos los grupos son de tamaño uno. En cada iteración, cada grupo es fusionado con otro grupo. Por lo tanto, el número de grupos al final de cada iteración es la mitad del número de grupos que había al inicio de la iteración. Este proceso finaliza cuando se llega a la solución trivial en la que todos los vértices se encuentran en un único grupo. La solución final es la solución con mayor calidad de entre las soluciones obtenidas al final de cada iteración.

IV. Implementación eficiente de la función objetivo

Para mejorar la eficiencia de las heurísticas propuestas para resolver el problema, es importante realizar una implementación eficiente del cálculo de la función objetivo. En



(a) Solución 1.

(b) Solución 2, obtenida tras mover el vértice v_1 al grupo C_2.

Figura 1: Evaluación de dos soluciones para el SMCP.

este trabajo, como se indica en la Sección II, la función objetivo utilizada es MQ, descrita inicialmente en [8]. Esta función describe la calidad de la modularidad de una solución atendiendo al acoplamiento y a la cohesión. El valor de la función objetivo debe ser maximizado. Es decir, cuanto mayor sea el valor de la función objetivo, mejor es la solución. En primer lugar, se calcula un factor de modularidad (MF, del inglés *Modularization Factor*) por cada grupo k, denominado MF_k. Este factor se calcula de la siguiente manera:

$$MF_k = \begin{cases} 0, & \text{if } i = 0 \\ \frac{i}{i+\frac{1}{2}j}, & \text{if } i > 0 \end{cases}$$

donde i es el número de aristas entre vértices del grupo k y j es el número de aristas entre un vértice del grupo k y un vértice de otro grupo. Finalmente, la calidad de la solución (MQ) se obtiene mediante el sumatorio de los factores de modularidad (MF) de cada grupo del grafo. Dada la definición de MF, la calidad de una solución se calcula como sigue:

$$MQ = \sum_{k=1}^{n} MF_k$$

donde n es el número total de grupos. Dado que la función objetivo es un sumatorio de valores que dependen de cada grupo de manera independiente, no es necesario calcularla cada vez que se produzca un cambio parcial en la solución. Para mover un vértice v_1 del grupo C_1 al grupo C_2, basta con actualizar MF_1 y MF_2 y aplicar la diferencia de los factores actualizados al valor MQ de la solución.

En la Figura 1, se muestra un ejemplo de un cambio parcial de una solución. En particular, mostramos el movimiento de un vértice (v_1) del grupo C_1 al grupo C_2. En la Figura 1a, se muestra el estado inicial de la solución. En la Figura 1b, se muestra el estado de la solución tras realizar el movimiento. En ambas figuras, se indica el factor de modularidad (MF) de cada grupo. Para calcular el valor de la función objetivo en el estado inicial ($MQ = 2,0$), se calcula el factor de modularidad de cada uno de los grupos y se suman sus valores. Tras realizar el movimiento de v_1, la función objetivo de la solución resultante se puede calcular de la siguiente manera:

1. Se actualiza el valor anterior de la función objetivo $(2,0)$ para no tener en cuenta los grupos que se ven afectados por el movimiento.
$$MQ = MQ - MF_1 - MF_2 = 1,0.$$
2. Se recalcula el factor de modularidad del grupo de origen ($MF_1 = 0,5$) y del grupo de destino ($MF_2 = 0,6666$) del vértice (v_1) que se ha movido.
3. Se actualiza el valor de MQ con los nuevos factores de modularidad de los grupos que han sufrido modificaciones.
$$MQ = MQ + MF_1 + MF_2 = 2,1666.$$

De esta manera, se evita evaluar la función objetivo para obtener la calidad de la solución resultante, recalculando únicamente los factores de modularidad de los dos grupos afectados.

V. Preprocesado de las instancias

Para reducir el tamaño de las instancias en el SMCP, Köhler et al. [21] propusieron un procedimiento de preprocesado. Este procedimiento se basa en que, en la solución óptima, un vértice que solo tiene una arista estará siempre en el mismo grupo que el vértice al que le une dicha arista. Para aplicar este procedimiento, primero hay que transformar el MDG en un grafo ponderado no dirigido. El procedimiento es el siguiente:

1. Se asigna un peso unitario a todas las aristas del grafo para convertirlo en un grafo ponderado.
2. Se elimina la dirección de las aristas, obteniendo un grafo no dirigido.
3. Se fusionan las aristas que unen el mismo par de vértices en una sola arista. La arista resultante tandrá un peso igual a la suma de los pesos de las aristas fusionadas.

4. Por cada vértice u con una sola arista que lo une a otro vértice v, se elimina el vértice u y se añade una arista (con el mismo peso que la arista que unía ambos vértices) que une al vértice v consigo mismo.

Los algoritmos planteados en la Sección III se pueden ejecutar sobre las instancias reducidas resultantes. Como se reporta en [19], la calidad de las soluciones sobre el grafo reducido es igual que la calidad de las soluciones sobre el grafo original. Finalmente, las soluciones obtenidas se pueden transformar en soluciones a las instancias originales aplicando el procedimiento inverso al preprocesado.

VI. Resultados

Para evaluar las heurísticas descritas en la Sección III, se ha seleccionado un conjunto de 47 proyectos de *software* que han sido utilizados previamente como instancias en otros trabajos relacionados y que se encuentran disponibles públicamente en [19]. Las instancias utilizadas varían en tamaño, teniendo una media de 269,17 vértices y 1328,49 aristas. La instancia más pequeña tiene 15 vértices y 64 aristas, mientras que la instancia de mayor tamaño tiene 1161 vértices y 5770 aristas. En la Tabla I, se reportan las instancias utilizadas en la experimentación y el número de vértices y aristas de cada instancia, antes (original) y después (reducido) del preprocesado. Además, se reportan los resultados obtenidos con las heurísticas propuestas. En particular, se reporta la calidad de la solución para cada instancia encontrada por cada algoritmo y el tiempo (en segundos) de ejecución por algoritmo e instancia. Todos los experimentos se han ejecutado en un equipo con un procesador AMD EPYC 7282 @ 2795 Mhz con 8 Cores y con 8 GB de RAM. El sistema operativo utilizado ha sido Microsoft Windows 10 Pro 10.0.19042 x64. Todos los métodos han sido programados en Java OpenJDK 15.0.2.

En la Tabla II, se comparan los resultados mencionados anteriormente. Por cada método, se reporta la calidad media de las soluciones obtenidas (MQ); el número de instancias para las que la calidad de la solución encontrada ha sido mayor que la calidad de las soluciones encontradas por el resto de métodos (Número de mejores soluciones); la desviación media, en términos de calidad, de las soluciones obtenidas con respecto a las mejores soluciones encontradas para cada instancia por todos los métodos (Desviación media (%)); y el tiempo medio de ejecución (CPUt (s)) en segundos. Como se puede observar en la tabla, las construcciones aleatorias (C1) ofrecen soluciones con una calidad menor a la de otros métodos más voraces. Además, ofrecen menor calidad que las soluciones iniciales de las instancias. Sin embargo, el resto de métodos ofrecen soluciones con mayor modularidad que las soluciones iniciales. Las soluciones encontradas por el algoritmo C4 tienen mayor calidad media que las encontradas por el algoritmo C3. Sin embargo, el tiempo de ejecución para el algoritmo C4 es considerablemente mayor. Finalmente, las soluciones obtenidas mediante el algoritmo C2 tienen mayor calidad que las soluciones obtenidas con el resto de algoritmos para todas las instancias. Además, la calidad media de las

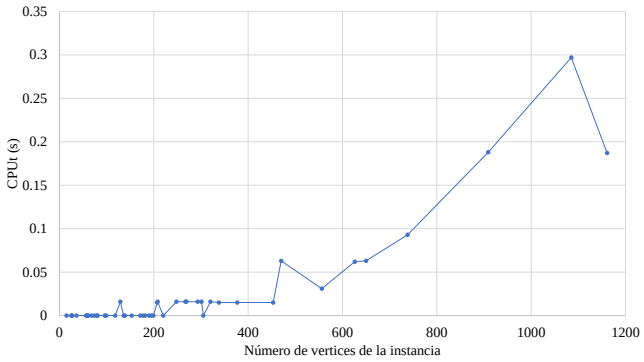


Figura 2: Tiempo de ejecución (en segundos) del algoritmo C2 con respecto al tamaño (en número de vértices) de cada instancia.

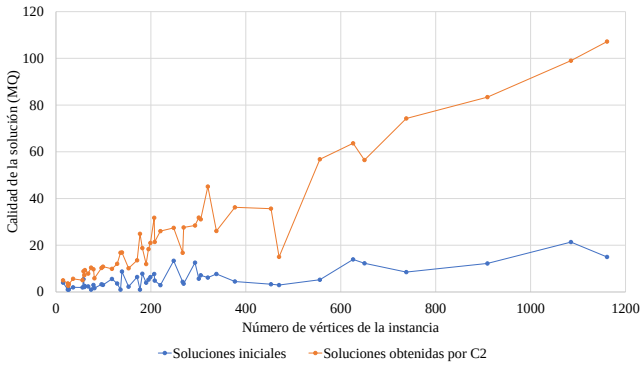


Figura 3: Comparación de la calidad de las soluciones obtenidas con el algoritmo C2 y las soluciones iniciales.

soluciones del método C2 es significativamente superior a la del resto de algoritmos. Finalmente, se puede observar que el tiempo de ejecución del algoritmo C2 es únicamente superado por las construcciones aleatorias (C1).

En la Figura 2, se representa el tiempo de ejecución del algoritmo C2, para cada instancia, según el tamaño (en número de vértices) de la instancia. Aunque se puede observar un crecimiento lineal del tiempo de ejecución conforme aumenta el tamaño de la instancia, el tiempo de ejecución para las instancias más grandes no llega a alcanzar la cota de las tres décimas de segundo.

En la Figura 3, se representa la calidad de las soluciones según el tamaño (en número de vértices) de la instancia. En color naranja, se representa la calidad de las soluciones obtenidas por el algoritmo C2. En color azul, se representa la calidad de las soluciones iniciales de cada instancia. Es decir, la calidad inicial del sistema *software* evaluado, antes de ser reorganizado por el algoritmo. Como se puede observar, existe una mejora significativa de la modularidad del sistema al aplicar el algoritmo C2. Además, esta mejora es aún mayor conforme crece el tamaño del sistema. De media, la calidad de las soluciones obtenidas por el algoritmo C2 es un 71,71 % mejor que la calidad de las soluciones iniciales. Esta mejora se

Tabla I: Resultados detallados de los algoritmos evaluados.

	Original		Reducido		MQ	C1		C2		C3		C4	
Instancia	$\|V\|$	$\|A\|$	$\|V\|$	$\|A\|$	inicial	MQ	CPUt (s)	MQ	CPUt (s)	MQ	CPUt (s)	MQ	CPUt (s)
jstl	15	20	14	15	4,0000	1,5714	<0,001	5,0000	<0,001	2,0428	<0,001	4,5714	<0,001
nanoxml	25	64	23	62	1,0000	0,9222	<0,001	3,6467	<0,001	2,8879	<0,001	3,3251	<0,001
jodamoney	26	102	26	85	1,5947	0,0714	<0,001	2,6480	<0,001	1,6961	<0,001	2,2210	<0,001
jxlsreader	27	73	25	73	1,0000	0,9791	<0,001	3,3333	<0,001	1,3384	<0,001	2,9546	<0,001
apache_zip	36	86	30	74	2,0000	1,9746	<0,001	5,6346	<0,001	3,2286	<0,001	4,8559	<0,001
udt-java	56	227	54	210	1,9411	0,5440	<0,001	5,0497	<0,001	3,2080	<0,001	3,9585	0,015
javaocr	58	155	43	126	5,4668	3,7727	<0,001	8,8741	<0,001	6,6322	<0,001	7,1466	<0,001
pfcda_base	60	197	54	168	2,1281	0,9605	<0,001	7,2129	<0,001	3,9749	<0,001	6,0600	<0,001
servletapi	61	131	47	122	2,5455	3,7129	<0,001	9,3856	<0,001	6,5884	<0,001	7,3578	<0,001
forms	68	270	64	224	2,3559	0,6501	<0,001	7,8973	<0,001	4,0839	<0,001	6,5624	<0,001
jscatterplot	74	232	68	171	1,0000	1,0093	<0,001	10,4107	<0,001	5,3239	<0,001	7,5586	<0,001
jxlscore	79	330	67	305	3,0152	3,9782	<0,001	9,7400	<0,001	6,3575	<0,001	7,3465	<0,001
jfluid	81	315	75	276	1,6461	1,0260	<0,001	5,8426	<0,001	3,2444	<0,001	5,1296	<0,001
jpassword	96	361	87	332	3,2753	2,5487	<0,001	10,2869	<0,001	5,7313	<0,001	8,4578	<0,001
junit	99	276	61	193	3,0259	4,8040	<0,001	10,8529	<0,001	6,7074	<0,001	8,7122	<0,001
xmldom	118	209	67	159	5,5833	4,1455	<0,001	9,91623	<0,001	7,9355	<0,001	8,4711	<0,001
tinytim	129	564	122	499	3,6161	1,6799	<0,001	12,0559	0,016	6,6736	<0,001	9,0305	0,031
jkaryoscope	136	460	127	345	1,0000	1,8791	<0,001	16,8152	<0,001	9,2614	<0,001	12,3835	0,031
gae_plugin_core	139	375	87	259	8,7638	6,7040	<0,001	16,9290	<0,001	11,4844	<0,001	13,7713	0,016
javacc	153	722	145	663	2,2775	2,1263	<0,001	10,1562	<0,001	5,6924	<0,001	7,8595	0,031
JavaGeom	171	1445	160	1339	6,4262	3,0179	<0,001	13,5802	<0,001	6,3184	<0,001	9,7571	0,063
jdendogram	177	583	148	447	1,0000	4,3702	0,015	24,8986	<0,001	12,3041	<0,001	19,3415	0,032
xmlapi	182	413	125	358	7,8174	4,9130	<0,001	18,7857	<0,001	12,6396	<0,001	15,1754	0,016
jmetal	190	1137	178	1123	3,9764	2,1059	<0,001	11,9713	<0,001	6,4665	<0,001	9,4330	0,062
dom4j	195	930	181	876	5,3055	3,5787	<0,001	18,3587	<0,001	10,2491	<0,001	14,6573	0,047
pdf_renderer	199	629	172	497	6,3812	5,4934	<0,001	20,9896	<0,001	11,4827	<0,001	16,1085	0,047
Jung_graph_model	207	603	187	532	7,6969	5,6995	<0,001	31,7812	0,015	14,7093	<0,001	19,2617	0,047
jung_visualization	208	919	192	837	4,8789	2,2427	<0,001	21,4228	0,016	12,2899	<0,001	16,1214	0,063
jconsole	220	859	187	663	2,9185	6,1193	<0,001	26,0489	<0,001	11,8618	<0,001	19,3080	0,062
pfcda_swing	248	885	237	801	13,3544	2,5018	<0,001	27,4341	0,016	16,9269	<0,001	20,6809	0,109
jml-1,0b4	267	1745	262	1671	4,3299	0,9394	<0,001	16,7586	0,016	8,4502	<0,001	12,3194	0,156
jpassword2	269	1348	248	1171	3,5410	3,6799	<0,001	27,6183	0,016	13,9026	0,016	21,2662	0,125
notelab-full	293	1349	273	1233	12,5691	3,4155	<0,001	28,4646	0,016	16,8398	0,016	22,8309	0,141
Poormans_CMS	301	1118	253	1009	5,6807	8,2830	<0,001	31,8417	0,016	19,8148	<0,001	23,7345	0,125
log4j	305	1078	258	944	7,1514	8,6761	<0,001	31,0967	<0,001	18,9189	<0,001	23,8619	0,125
jtreeview	320	1057	268	830	6,1688	12,8270	<0,001	45,1377	0,016	26,2366	<0,001	32,0857	0,156
JACE	338	1524	271	1209	7,7158	7,0101	<0,001	26,1157	0,015	13,7999	0,015	18,2311	0,156
javaws	377	1403	293	1107	4,4952	8,6419	<0,001	36,2345	0,015	19,4290	0,016	27,0381	0,187
lwjgl-2,8,4	453	1976	392	1790	3,3185	7,5041	<0,001	35,6589	0,015	19,3885	0,015	23,4744	0,375
res_cobol	470	7163	461	5690	2,9695	1,1572	0,016	15,0121	0,063	6,5478	0,016	8,7233	0,781
y_base	556	2510	479	2166	5,2471	10,1458	<0,001	56,7902	0,031	27,1972	0,031	40,8491	0,657
apache_ant_taskdef	626	2421	538	2108	13,9474	14,1433	<0,001	63,6499	0,062	37,2615	0,031	48,0098	0,937
itextpdf	650	3898	603	3555	12,2911	7,8630	0,015	56,4792	0,063	29,3401	0,032	41,0507	1,328
apache_lucene_core	738	3726	712	3330	8,5130	6,4436	<0,001	74,2559	0,093	38,0147	0,063	55,4891	1,922
eclipse_jgit	909	5452	872	4904	12,2163	7,4765	0,015	83,4062	0,188	46,5513	0,109	62,0638	3,609
apache_ant	1085	5329	999	4881	21,3681	11,8097	0,046	99,0074	0,297	53,0630	0,531	72,7795	5,328
ylayout	1161	5770	1004	4990	14,9800	21,8091	<0,001	107,1737	0,187	57,0754	0,172	77,3803	5,125

ve incrementada a un 83,28 % de media para las 5 instancias de mayor tamaño.

VII. Conclusiones

En este trabajo, se han evaluado 4 heurísticas para la modularización de sistemas *software*, atendiendo a la calidad de las soluciones obtenidas y al tiempo necesario para obtener dichas soluciones. Para la evaluación de la calidad, se ha utilizado una métrica aceptada por la comunidad, MQ, que mide la modularidad de las soluciones teniendo en cuenta sus niveles de acoplamiento y cohesión. Además, se ha presentado una implementación eficiente para medir el MQ de las soluciones al SMCP. Las heurísticas planteadas se han evaluado con un conjunto de instancias propuestas en el estado del arte y que están disponibles públicamente. Estas instancias, además, se corresponden con proyectos *software* reales. Sobre este conjunto de instancias, se ha mostrado que la calidad de las soluciones obtenidas por las heurísticas descritas es mayor que la calidad de las soluciones iniciales. Es decir, los algoritmos planteados mejoran la mantenibilidad de los proyectos *software* evaluados. Además, esta diferencia es más significativa cuanto mayor es el tamaño del sistema. Los algoritmos planteados se muestran prometedores para mejorar la modularidad y, por lo tanto, la mantenibilidad de los proyectos *software*. Finalmente, la evaluación de las 4 heurísticas planteadas en este trabajo puede servir de base para plantear, posteriormente, algoritmos constructivos en metaheurísticas.

Referencias

[1] "ISO/IEC/IEEE International Standard - Systems and software engineering – Software life cycle processes," *ISO/IEC/IEEE 12207:2017(E) First edition 2017-11*, pp. 1–157, 2017.

[2] T. Bakota, P. Hegedus, G. Ladanyi, P. Kortvelyesi, R. Ferenc, and T. Gyimothy, "A cost model based on software

Tabla II: Comparación entre los diferentes algoritmos propuestos.

Método	MQ	Número de mejores soluciones	Desviación media (%)	CPUt (s)
C1	4,83	0	78,94 %	<0,01
C2	25,99	47	0,00 %	0,02
C3	14,28	0	43,47 %	0,02
C4	19,34	0	22,89 %	0,47
Soluciones iniciales	5,66	0	71,45 %	0.00

maintainability," in *IEEE International Conference on Software Maintenance, ICSM*, pp. 316–325, 2012.

[3] C. Chen, R. Alfayez, K. Srisopha, B. Boehm, and L. Shi, "Why is it important to measure maintainability and what are the best ways to do it?," in *Proceedings - 2017 IEEE/ACM 39th International Conference on Software Engineering Companion, ICSE-C 2017*, pp. 377–378, Institute of Electrical and Electronics Engineers Inc., jun 2017.

[4] S. McConnell, *Code complete*. Pearson Education, 2004.

[5] C. Larman, *Applying UML and patterns: an introduction to object oriented analysis and design and interative development*. Pearson Education India, 2012.

[6] S. Gibbs, E. Casais, O. Nierstrasz, X. Pintado, and D. Tsichritzis, "Class management for software communities," *Communications of the ACM*, vol. 33, no. 9, pp. 90–103, 1990.

[7] L. C. Briand, S. Morasca, and V. R. Basili, "Defining and validating measures for object-based high-level design," *IEEE Transactions on Software Engineering*, vol. 25, no. 5, pp. 722–743, 1999.

[8] S. Mancoridis, B. S. Mitchell, C. Rorres, Y.-F. R. Chen, and E. R. Gansner, "Using automatic clustering to produce high-level system organizations of source code," in *Proceedings. 6th International Workshop on Program Comprehension. IWPC'98 (Cat. No. 98TB100242)*, pp. 45–52, IEEE, 1998.

[9] M. R. Garey and D. S. Johnson, *Computers and Intractability; A Guide to the Theory of NP-Completeness*. USA: W. H. Freeman & Co., 1990.

[10] U. Brandes, D. Delling, M. Gaertler, R. Gorke, M. Hoefer, Z. Nikoloski, and D. Wagner, "On Modularity Clustering," *IEEE transactions on knowledge and data engineering*, vol. 20, pp. 172–188, feb 2008.

[11] K. Praditwong, "Solving software module clustering problem by evolutionary algorithms," in *2011 Eighth International Joint Conference on Computer Science and Software Engineering (JCSSE)*, pp. 154–159, IEEE, 2011.

[12] J. Huang, J. Liu, and X. Yao, "A multi-agent evolutionary algorithm for software module clustering problems," *Soft Computing*, vol. 21, no. 12, pp. 3415–3428, 2017.

[13] A. Prajapati and J. K. Chhabra, "A particle swarm optimization-based heuristic for software module clustering problem," *Arabian Journal for Science and Engineering*, vol. 43, no. 12, pp. 7083–7094, 2018.

[14] A. Prajapati and J. K. Chhabra, "Harmony search based remodularization for object-oriented software systems," *Computer Languages, Systems & Structures*, vol. 47, pp. 153–169, 2017.

[15] A. Prajapati and J. K. Chhabra, "MaDHS: Many-objective discrete harmony search to improve existing package design," *Computational Intelligence*, vol. 35, no. 1, pp. 98–123, 2019.

[16] A. Prajapati and J. K. Chhabra, "Many-objective artificial bee colony algorithm for large-scale software module clustering problem," *Soft Computing*, vol. 22, no. 19, pp. 6341–6361, 2018.

[17] B. G. Varghese R., K. Raimond, and J. Lovesum, "A novel approach for automatic remodularization of software systems using extended ant colony optimization algorithm," *Information and Software Technology*, vol. 114, pp. 107–120, oct 2019.

[18] A. F. Pinto, A. C. de Faria Alvim, and M. de Oliveira Barros, "ILS for the Software Module Clustering Problem," *XLVI Simpósio Brasileiro de Pesquisa Operacional. Salvador:[sn]*, pp. 1972–1983, 2014.

[19] M. C. Monçores, A. C. F. Alvim, and M. O. Barros, "Large neighborhood search applied to the software module clustering problem," *Computers & Operations Research*, vol. 91, pp. 92–111, 2018.

[20] G. Karypis and V. Kumar, "A Fast and High Quality Multilevel Scheme for Partitioning Irregular Graphs," *SIAM J. Sci. Comput.*, vol. 20, pp. 359–392, dec 1998.

[21] V. Köhler, M. Fampa, and O. Araújo, "Mixed-integer linear programming formulations for the software clustering problem," *Computational Optimization and Applications*, vol. 55, no. 1, pp. 113–135, 2013.

Un algoritmo eficiente para el problema de disposición de instalaciones en dos filas

Raúl Martín-Santamaría, Alberto Herrán, Abraham Duarte, J. Manuel Colmenar
Universidad Rey Juan Carlos
Móstoles, Spain
{raul.martin, alberto.herran, abraham.duarte, josemanuel.colmenar}@urjc.es

***Resumen*—El problema de disposición en dos filas, DRLP (del inglés, *Double Row Layout Problem*), es un problema muy relacionado con el ámbito de los almacenes y cadenas de montaje, donde existe un pasillo central que da acceso a diferentes elementos a ambos lados del pasillo. Su objetivo es establecer una disposición de instalaciones de modo que se minimice el coste total, definido como la suma de distancias horizontales entre cada par de instalaciones, multiplicado por un coste fijo, que viene a representar el flujo entre ambas instalaciones. Se trata de un problema NP duro, por lo que se propone un algoritmo metaheurístico GRASP capaz de generar buenas soluciones para instancias de tamaño moderado en tiempos muy contenidos. La propuesta se comparará con el estado del arte, destacando la eficiencia para generar resultados competitivos.**

***Index Terms*—Ubicación de instalaciones, heurísticas, GRASP, metaheurísticas**

I. Introducción

Los problemas de disposición de instalaciones, abreviados como FLP (del inglés, *Facility Layout Problems*), forman una familia cuya característica común es la necesidad de disponer o colocar una serie de instalaciones atendiendo a diversas restricciones. Dentro de esta familia, dependiendo de las restricciones o el objetivo que se persiga, se tienen diferentes problemas [13]. Por ejemplo, si la restricción principal es el número de filas disponibles, se tiene el problema de disposición en una fila, denominado SRLP (*Single Row Layout Problem*) [17], en dos filas, denominado DRLP (*Double Row Layout Problem*) [10], y en tres o más filas, denominados MRFLP (*Multi Row FLP*) [14]. Si existen restricciones adicionales como que no pueda haber espacios entre las instalaciones y que todas estén alineadas a la izquierda, se tiene, por ejemplo, el problema denominado CAP (*Corridor Allocation Problem*) [3], que tiene en cuenta también dos filas. Si las instalaciones no pueden cambiar de fila, se trata de los problemas denominados PROP (*Parallel Row Ordering Problem*) [5]. Además de las variantes mencionadas, existen algunas otras definidas a partir de otros tipos de restricciones.

Este trabajo se centra, en concreto, en el DRLP, problema considerado como NP duro [7], cuyo objetivo es disponer una serie de instalaciones o máquinas en dos filas paralelas de manera que se minimice una función de coste que considera la distancia entre los puntos centrales de cada máquina, ponderada por pesos asociados a cada pareja de máquinas. Esta ponderación tiene como objetivo modelar el traslado de material de una máquina a otra, también llamado flujo entre instalaciones. De esta manera, dos máquinas que no comparten material tendrán peso cero, de modo que podrán estar alejadas entre sí. Por el contrario, dos máquinas con una relación estrecha tendrán un valor de peso elevado, de modo que será necesario situarlas lo más próximas posible entre sí para reducir el valor final de la función objetivo.

Como restricciones adicionales se definen las siguientes: (1) las máquinas o instalaciones tienen diferentes anchuras, considerando su altura (desde el punto de vista del plano) como no relevante para el problema; (2) las máquinas se pueden colocar en cualquiera de las dos filas; (3) puede haber espacios libres entre las máquinas; (4) no es necesario alinear las máquinas sobre ninguna referencia.

Los problemas de disposición de instalaciones tienen diversas aplicaciones prácticas en el mundo industrial, destacando el ámbito de la fabricación y cadenas de montaje. Como ejemplo de aplicación del DRLP, se puede tomar el caso de una fábrica donde un robot situado entre las dos filas se encarga del traslado de material entre máquinas [12]. Este robot sería capaz de recuperar elementos de ambas filas y moverlos entre máquinas. El objetivo, por tanto, sería minimizar el tiempo de ensamblado total, que depende de la distancia que debe recorrer horizontalmente el robot para desplazarse, multiplicado por el número de veces que debe realizar cierta operación o flujo entre ambas instalaciones. La Figura 1 muestra este ejemplo de manera gráfica.

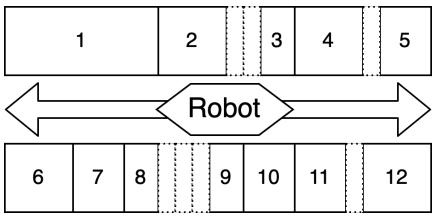


Figura 1. Ejemplo de cadena de montaje con dos filas de máquinas y un robot central.

En la literatura se pueden encontrar más aplicaciones del DRLP como la optimización de líneas de producción [10] o la fabricación de semiconductores [18].

Este problema se ha trabajado, fundamentalmente, a través de algoritmos exactos. Chung y Tanchoco proponen un modelo con programación entera mixta (MIP) que resuelven con CPLEX en el primer artículo donde se define este problema

[10]. Para acelerar el proceso, en ese trabajo se proponen cinco heurísticas que generan soluciones de partida que sirven como cota superior. Más adelante, Amaral ha ido proponiendo de manera sucesiva diferentes modelos MIP a lo largo de los años [4], [6], [8], hasta llegar a proponer un enfoque de dos fases donde un modelo MIP se combina con diferentes métodos heurísticos, consiguiendo buenos resultados y permitiendo atacar instancias más grandes [7].

El objetivo del trabajo que se presenta en este artículo es proponer un algoritmo eficiente para resolver el DRLP empleando tiempos de cómputo reducidos. Para ello se propone la utilización de un algoritmo GRASP con un diseño sencillo que permita atacar instancias grandes empleando tiempos muy reducidos. En la comparativa de esta propuesta con el estado del arte se consiguen resultados muy competitivos, llegando al 0.13 % del coste obtenido por el algoritmo previo, pero empleando tiempos de ejecución tres órdenes de magnitud inferiores.

El resto del artículo se organiza de la siguiente manera. En la Sección II se describe formalmente el problema. La propuesta algorítmica y los resultados experimentales se presentan en las secciones III y IV. Finalmente, las conclusiones y el trabajo futuro se detallan en la Sección V.

II. Descripción formal del problema

Para facilitar la comparación con el estado del arte, se utilizará la descripción formal del problema siguiendo la notación introducida en [7], que propone las siguientes definiciones:

- n: Número de instalaciones o elementos a colocar.
- N: Conjunto de instalaciones o elementos a colocar. $N = \{1, 2, ..., n\}$
- R: Conjunto de filas, $R = \{1, 2\}$
- l_i: Anchura de la instalación $i \in N, l_i \in \mathbb{N}$
- c_{ij}: Flujo entre la instalación i y la instalación j, con $i, j \in N$. Se trata de un dato de la propia instancia.
- φ: Disposición de instalaciones donde, para cada una de ellas se determina la fila en que se ubica, así como la posición horizontal donde se sitúa.
- d_{ij}^{φ}: Distancia horizontal entre los puntos centrales de la instalación i y la instalación j, con $i, j \in N$, según la disposición φ.

Además de estas, se utilizará φ_i representando el conjunto de instancias ubicadas en la fila i en la solución φ.

El problema, por tanto, consiste en crear una disposición de instalaciones φ utilizando las dos filas disponibles, de forma que se minimice la función objetivo $\mathcal{F}$ que se muestra en la Ecuación (1).

$$\mathcal{F}(\varphi) = \sum_{1 \leq i < j \leq n} c_{ij} d_{ij}^{\varphi} \quad (1)$$

Como se ha mencionado anteriormente, es importante destacar que la distancia vertical entre ambas filas se considera despreciable. Por tanto, solamente se tiene en cuenta la distancia horizontal para el cálculo de la función de objetivo.

En este problema existen infinitas combinaciones para la disposición de las instalaciones, puesto que se pueden colocar en cualquier posición siempre que no solapen con otras instalaciones en la misma fila. Además puede haber huecos de tamaño arbitrario entre instalaciones de la misma fila. Para modelar esta situación, se propone el uso de instalaciones *fantasma*. Estas instalaciones no forman parte de la instancia original, y carecen de flujo con el resto de instalaciones, lo que supone que nunca incrementarán por sí mismas el coste de la función objetivo. Su utilidad radica en transformar el problema en la búsqueda de una secuencia de permutaciones, de forma que no existen huecos arbitrarios entre instalaciones ya que las instalaciones de la instancia original pueden estar separadas por instalaciones *fantasma*. La Figura 1 muestra con líneas discontinuas las instalaciones *fantasma*.

III. Propuesta algorítmica

Para atacar este problema se propone utilizar la metaheurística GRASP (del inglés, *Greedy Randomized Adaptative Search Procedure*, procedimiento de búsqueda adaptativo voraz aleatorizado) [11]. Este procedimiento consiste en la utilización de un método constructivo voraz aleatorizado seguido por una fase de búsqueda local. Como la calidad de las soluciones va muy ligada a la solución inicial del constructivo, se utilizará un enfoque multi-arranque. En los siguientes apartados se describe tanto el enfoque GRASP como los métodos constructivos y búsquedas locales propuestas en el trabajo.

III-A. Esquema GRASP

El pseudocódigo del algoritmo GRASP propuesto en este trabajo se muestra en el Algoritmo 1.

Como se puede ver, el método GRASP recibe dos parámetros: $maxIter$, que indica el número máximo de iteraciones que se realizarán en este enfoque multi-arranque, y α, que es el parámetro que equilibra el grado de aleatoriedad en el método constructivo. Los dos primeros pasos del método GRASP inicializan tanto el número de iteraciones, $iter$, como la mejor solución, φ_{best}. A continuación se inicia la ejecución del bucle principal, que comienza con la construcción de una solución φ en el paso 4 invocando uno de los métodos constructivos que se describen en la siguiente sección. Seguidamente se ejecuta una búsqueda local sobre la solución construida que da como resultado una nueva solución φ'. Si esta solución tiene menor valor de función objetivo que la mejor solución generada hasta el momento, esta se actualiza (pasos 6 y 7). A continuación, se incrementa el número de iteraciones de manera que el bucle continúa hasta llegar al máximo. La mejor solución obtenida, φ_{best}, se devuelve en el paso 9.

III-B. Métodos Constructivos

En este artículo se proponen dos enfoques constructivos tipo GRASP. Por un lado se propone la variante clásica, que se denominará $C1$, donde primero interviene la parte voraz y, seguidamente, la parte aleatoria. Por otro lado se propone una variante inversa, que se denominará $C2$, donde primero se realiza una selección aleatoria y, a continuación, se aplica el criterio voraz.

Algoritmo 1: *GRASP*$(maxIter, \alpha)$

1 $iter = 0$
2 $\varphi_{best} \leftarrow \emptyset$
3 **while** $iter < maxIter$ **do**
4 $\quad \varphi \leftarrow Constructivo(\alpha)$
5 $\quad \varphi' \leftarrow BusquedaLocal(\varphi)$
6 $\quad$ **if** $\mathcal{F}(\varphi') < \mathcal{F}(\varphi_{best})$ **then**
7 $\quad\quad \varphi_{best} \leftarrow \varphi'$
8 $\quad iter \leftarrow iter + 1$
9 **return** φ_{best}

Algoritmo 2: C1 (α)

1 $\varphi \leftarrow \emptyset$
2 $CL \leftarrow$ `CreaPosicionesIniciales`(N)
3 **while** $|CL| > 0$ **do**
4 $\quad g_{min} = \min_{(u,i,j)\in CL} g(\varphi, u, i, j)$
5 $\quad g_{max} = \max_{(u,i,j)\in CL} g(\varphi, u, i, j)$
6 $\quad RCL \leftarrow \{(u,i,j) \in CL : g(\varphi, u, i, j) \leq g_{min} + \alpha \cdot (g_{max} - g_{min})\}$
7 $\quad (u,i,j) \leftarrow$ `SeleccionarAleatorio`(RCL)
8 $\quad \varphi \leftarrow \varphi \cup (u,i,j)$
9 $\quad CL \leftarrow$ `ActualizaPosiciones`(CL, u, i, j)
10 **return** φ

En ambos métodos constructivos se utilizará un enfoque incremental, en el que, partiendo de una solución vacía, se irá asignando en cada iteración una instalación en la posición apropiada, dependiendo de la estrategia seguida. En ambos casos, la función voraz $g(\varphi, u, i, j)$ determinará el coste de añadir la instalación u en la posición j de la fila i a la solución dada φ.

El Algoritmo 2 muestra el pseudocódigo del método $C1$, donde se comienza con la parte voraz y se continúa con la aleatoria. En el primer paso se crea una solución vacía φ para, seguidamente, inicializar la lista de candidatos CL incluyendo en ella a todas las posibles posiciones para las instalaciones del problema (paso 2). Estas posiciones candidatas son tripletas con la forma (u, i, j) donde u es la instalación, i es la fila y j es la posible posición para cada instalación. Dado que se trata del paso inicial, las instalaciones se podrán colocar únicamente en la posición 1 de las filas 1 y 2. El bucle principal comienza en el paso 3, y terminará cuando todas las instalaciones hayan sido asignadas, dejando la lista de candidatos vacía. En cada iteración se calcula el coste mínimo y máximo de añadir una instalación en ambas filas en cualquier posición de las disponibles en CL (pasos 4 y 5). Utilizando ambos valores, se calcula el umbral de aceptación de la lista de candidatos restringida, RCL, de forma que todas las posibles asignaciones cuyo valor dado por la función g sea menor que este umbral se incluyen en RCL. Tal y como se ve en el paso 6, el valor de α determina el equilibrio entre aleatoriedad y voracidad. Si $\alpha = 0$, se trata de un enfoque totalmente voraz, mientras que $\alpha = 1$ es equivalente a un enfoque totalmente aleatorio. Una vez creada la RCL, se elige de manera aleatoria una de las posiciones y se incluye en la solución (pasos 7 y 8). En el último paso del bucle, la lista de candidatos se actualiza eliminando todas las posiciones correspondientes a la instalación u elegida previamente de la RCL, e incluyendo las nuevas posiciones posibles tras la inserción de esta nueva instalación. Finalmente, la solución φ se devuelve en el paso 10.

Por otro lado, en la segunda variante constructiva, denominada $C2$, se procede al revés. En primer lugar se genera una RCL seleccionando aleatoriamente un conjunto de posiciones de la CL. En este caso, el parámetro α pondera el número de elementos a elegir, que vendrá dado por la siguiente fórmula: $max(1, \lfloor(1-\alpha)\cdot|CL|\rfloor)$. A continuación, se escoge la posición de la RCL con mejor valor según la función voraz g. Cabe destacar que, de nuevo, α equilibra el grado de aleatoriedad. Como se aprecia en la fórmula anterior, el significado de α es el mismo que en $C1$ ya que $\alpha = 0$ se corresponde con un enfoque totalmente voraz, mientras que $\alpha = 1$ provoca un comportamiento totalmente aleatorio. El pseudocódigo de este método se omite por necesidad de espacio.

III-C. Búsqueda Local

En este trabajo se propone que el método de búsqueda local indicado en el paso 5 del Algoritmo 1 utilice la estrategia *best improvement*. En esta estrategia se recorre el vecindario de soluciones completo escogiendo la solución vecina con el mejor valor para la función objetivo.

Bajo este enfoque se proponen tres vecindarios de soluciones, generados por tres movimientos diferentes: intercambio, desplazamiento y *2-OPT*. A continuación se describe cada uno de los movimientos, así como los vecindarios que generan.

Intercambio: El movimiento de intercambio, consiste en intercambiar dos instalaciones, sean de la misma fila o de diferentes filas. En la Figura 2 se ilustra este comportamiento a través del intercambio de las instalaciones 3 y 11. Las instalaciones cuya posición podría cambiar se denotan en la figura usando fondo gris claro, mientras que las instalaciones *fantasma* se denotan con el borde discontinuo. Las soluciones a las que se puede llegar a través de este movimiento forman el vecindario $N_{intercambio}$.

Desplazamiento: El movimiento de desplazamiento consiste en el cambio de posición de una instalación, insertándola en cualquier otra posición, independientemente de la fila de origen o destino. La figura 3 muestra el ejemplo de desplazamiento de la instalación 3, denotando de nuevo con fondo gris claro las instalaciones cuya posición varía, y con línea discontinua las instalaciones *fantasma*. Las soluciones que se alcanzan a través de este movimiento forman el vecindario $N_{desplazamiento}$.

2-OPT: El movimiento 2-OPT que se propone está inspirado en el clásico movimiento de los problemas de rutas [15]. La propuesta consiste en elegir dos índices j y j' dentro

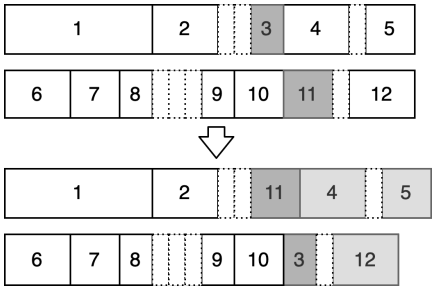


Figura 2. Intercambio entre las instalaciones 3 y 5. Se muestran en gris oscuro las instalaciones intercambiadas y en gris claro todas las instalaciones potencialmente desplazadas.

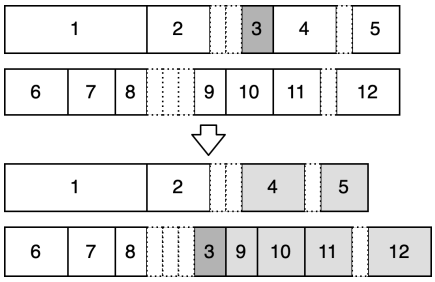


Figura 3. Movimiento de desplazamiento sobre la instalación 3. Se muestra en gris oscuro la instalación movida y en gris claro todas las instalaciones potencialmente desplazadas.

de una misma fila i, tales que $1 \leq j < j' + 1 < |\varphi_i|$, revirtiendo el orden de todas las instalaciones dentro del segmento seleccionado. La figura 4 muestra un ejemplo de aplicación de este movimiento en el segmento que engloban las instalaciones 6 y 11. El código de colores es similar al de las anteriores figuras. Las soluciones generadas por el movimiento $2-OPT$ forman el vecindario N_{2-OPT}.

IV. Experimentos

Esta sección comienza detallando los experimentos preliminares realizados, cuyo objetivo es determinar la configuración final del algoritmo. Esta configuración final se comparará con los resultados del estado del arte más recientes, publicados en [7]. La propuesta ha sido implementada utilizando Java

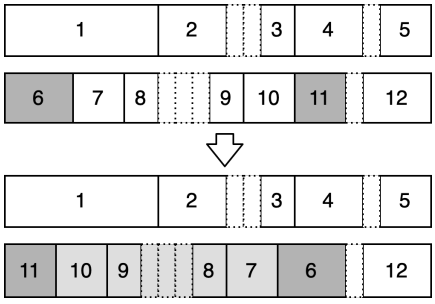


Figura 4. Movimiento 2-OPT aplicado sobre el segmento formado por las instalaciones 6 a 11. Se muestra en gris oscuro las instalaciones que limitan el segmento a modificar y en gris claro el resto de instalaciones desplazadas.

11, en una CPU Intel i7-8559U a 2.7GHz limitado a 4GB de RAM. En todos los experimentos que involucren alguno de los métodos de la propuesta GRASP se han utilizado 6 instancias *fantasma* de tamaño 0.5. Esta configuración se ha determinado tras una serie de experimentos previos.

Se han usado un total de 38 instancias existentes en la literatura previa. En concreto, las instancias *S9*, *S9H*, *S10* y *S11* provienen de [17], las instancias *Am** provienen de [6], [16] y por último *15* y *17* provienen de [1], [2]. Además, se han usado instancias de tamaño mayor: las instancias $n = 30$ introducidas en [9] y las instancias $n = 40$ propuestas en [7].

IV-A. Experimentos preliminares

Dado que se presentan dos métodos constructivos con un parámetro α para regular el grado de aleatoriedad y tres vecindarios para la búsqueda local, se propone una experimentación incremental que ayude a decidir la configuración más eficiente para el algoritmo final.

En primer lugar se muestra un experimento donde se comparan los dos métodos constructivos, $C1$ y $C2$, para los siguientes valores de su parámetro de entrada: $\alpha = \{0, 0{,}25, 0{,}5, 0{,}75, 1\}$. Además se añade el valor denotado como *Rnd*, que significa que se ha generado un valor aleatorio para α en cada iteración del bucle principal. En el caso de $\alpha = 0$, ambos métodos constructivos se comportan de manera voraz, por lo que se ha denotado su resultado como *Voraz*. De manera análoga, para $\alpha = 1$ se denota el resultado como *Aleatorio*.

La Tabla I muestra los resultados obtenidos tras ejecutar 100 iteraciones sobre el conjunto de instancias. Para cada método y valor de α se muestra el promedio de la desviación de los resultados con respecto al mejor valor obtenido durante el experimento (Dev. Pr. (%)), el tiempo de ejecución en segundos (T(s)) y el número de veces que se obtiene el mejor valor (#Best). Como se puede apreciar, el mejor resultado, tanto en desviación como en número de mejores soluciones obtenidas, lo consigue la configuración voraz. Los tiempos de ejecución muestran valores similares.

Tabla I
Resultados para diferentes configuraciones de los métodos constructivos.

Método	α	Dev. Pr. (%)	T(s)	#Best
Voraz	0	**0.97**	0.66	**11**
$C1$	0.25	10.91	0.64	0
	0.5	19.34	0.69	0
	0.75	27.15	0.73	0
	Rnd	24.73	0.79	0
$C2$	0.25	1.24	0.68	8
	0.5	2.26	0.67	6
	0.75	4.92	0.63	0
	Rnd	3.47	0.68	5
Aleatorio	1	33.86	0.76	0

El esquema GRASP involucra también un paso de búsqueda local, por lo que se ha repetido el mismo experimento pero

añadiendo, tras el método constructivo, una búsqueda local del tipo *best improvement* que explora los tres vecindarios descritos en la Sección III-C simultáneamente. Esta búsqueda se denomina *búsqueda local extendida*. La Tabla II muestra los resultados obtenidos siguiendo el mismo esquema de presentación que la tabla anterior. Tal y como se puede apreciar, los mejores resultados se obtienen en los casos de $C1$ con $\alpha = 0{,}25$ y $C2$ con $\alpha = 0{,}5$. Esto demuestra que la aleatoriedad introducida en los métodos $C1$ y $C2$ favorece el desempeño de la búsqueda local en comparación con la configuración totalmente voraz.

Tabla II
RESULTADOS PARA DIFERENTES CONFIGURACIONES DE LOS MÉTODOS CONSTRUCTIVOS SEGUIDOS DE UNA BÚSQUEDA LOCAL EXTENDIDA.

Método	α	**Dev. Pr. (%)**	**T(s)**	**#Best**
Voraz	0	0.12	**8.52**	18
$C1$	0.25	**0.08**	10.70	21
	0.5	0.14	11.59	18
	0.75	0.16	12.36	16
	RND	0.17	15.70	18
$C2$	0.25	0.10	8.71	17
	0.5	**0.08**	8.92	**22**
	0.75	0.09	9.34	18
	RND	0.15	10.48	15
Aleatorio	1	0.16	13.14	18

Llegados a este punto, es importante conocer el desempeño en el proceso de optimización de cada uno de los movimientos propuestos para este problema. Para tratar de medir ese impacto, se propone un nuevo experimento donde las dos configuraciones de método constructivo con mejor resultado en el experimento anterior se combinan con cuatro búsquedas locales que exploran, respectivamente, cada uno de los vecindarios individuales generados por los movimientos propuestos, así como también el vecindario extendido.

Tabla III
RESULTADOS DE LOS DOS MEJORES CONSTRUCTIVOS SEGUIDOS DE LAS DIFERENTES BÚSQUEDAS LOCALES PROPUESTAS.

Método	**Vecindario**	**Dev. Pr. (%)**	**T(s)**	**#Best**
$C1(0{,}25)$	*Extendido*	**0.03**	13.52	25
	N_{2-OPT}	1.70	2.16	0
	$N_{intercambio}$	0.23	8.64	10
	$N_{deplazamiento}$	1.35	5.72	2
$C2(0{,}5)$	*Extendido*	0.04	9.58	**26**
	N_{2-OPT}	1.51	**1.61**	3
	$N_{intercambio}$	0.13	5.93	17
	$N_{deplazamiento}$	1.29	4.18	3

A la vista de los resultados de este experimento, mostrados en la Tabla III, se puede observar que todos los vecindarios contribuyen al proceso de optimización, en mayor o menor medida. El vecindario individual con mejores resultados es $N_{intercambio}$. Sin embargo, es el vecindario extendido el que llega a los mejores valores de desviación promedio y número de mejores soluciones obtenidas en ambos constructivos. A pesar de que la diferencia en tiempo de ejecución es significativa en términos porcentuales, los valores absolutos son muy bajos. Por tanto, se determina que la configuración elegida para la comparativa final sea la formada por el constructivo $C2$ con $\alpha = 0{,}5$ dado que obtuvo el mayor número de mejores resultados empleando menos tiempo que la mejor configuración de $C1$.

IV-B. Comparativa con estado del arte

En esta sección se comparan los resultados obtenidos por la propuesta GRASP bajo la configuración descrita previamente contra la mejor heurística conocida para este problema, denotada como H2+LP [7]. Los tiempos de esa heurística se han multiplicado por diez, debido a que ejecutan el algoritmo diez veces, reportando la mejor solución, pero el tiempo reportado en su publicación corresponde con una sola iteración. En el caso de la propuesta GRASP de este trabajo, se reporta el tiempo total y mejor resultado obtenido tras 100 iteraciones del algoritmo.

La Tabla IV muestra los resultados de la comparativa utilizando el conjunto de instancias completo reportado en [7]. En particular, se muestra el mejor coste obtenido (***C.***), el tiempo total empleado en segundos (***T(s)***) y la desviación promedio respecto al mejor valor obtenido (***Dev. Pr. %)***).

Los resultados obtenidos por el enfoque GRASP son muy prometedores en términos de calidad, obteniendo el mejor resultado para 20 de las 24 instancias donde se conoce el óptimo. Además, obtiene una desviación promedio del 0.13 % en el conjunto total de instancias.

Por tanto, la principal contribución del algoritmo GRASP propuesto es su eficiencia, ya que la diferencia en tiempo de ejecución con el método previo es de tres órdenes de magnitud.

V. CONCLUSIONES

El DRLP es un problema NP-Duro, para el que, aunque existen varios modelos exactos, solo son capaces de resolver instancias de tamaño muy pequeño.

En este trabajo se proponen dos métodos constructivos y cuatro vecindarios diferentes dentro de un esquema algorítmico tipo GRASP. El objetivo principal de la propuesta es definir un algoritmo muy eficiente que sea capaz de obtener buenos resultados empleando tiempos de ejecución pequeños, permitiendo así atacar instancias grandes.

La comparativa de la propuesta GRASP con el método previo del estado del arte arroja resultados muy prometedores, alcanzando el mejor valor en 20 de las 24 instancias donde se conoce el óptimo y obteniendo un rendimiento promedio un 0.13 % inferior al previo. Estos resultados se consiguen empleando un tiempo promedio 1000 veces menor que el tiempo empleado por el algoritmo previo, lo que demuestra la eficiencia de la propuesta GRASP.

Como trabajo futuro se plantea la exploración de metaheurísticas similares que permitan mejorar los resultados de la literatura utilizando tiempos de ejecución contenidos.

Tabla IV
COMPARATIVA FINAL. LAS INSTANCIAS CON ÓPTIMO CONOCIDO SE INDICAN CON ⋆.

	H2+LP [7]			GRASP		
Instancia	***C.***	***T(s)***	***Dev. Pr. (%)***	***C.***	***T(s)***	***Dev. Pr. (%)***
S9⋆	1179.0	59.40	**0.0**	1179.0	0.18	**0.0**
S9H⋆	2293.0	58.20	**0.0**	2293.0	0.15	**0.0**
S10⋆	1351.0	90.60	**0.0**	1351.0	0.26	**0.0**
S11⋆	3424.5	135.00	**0.0**	3424.5	0.28	**0.0**
Am11a⋆	5559.5	132.70	0.01	5621.5	0.30	1.12
Am11b⋆	3655.5	137.90	**0.0**	3655.5	0.31	**0.0**
Am11c⋆	3832.5	133.90	**0.0**	3832.5	0.28	**0.0**
Am11d⋆	906.5	134.00	**0.0**	906.5	0.30	**0.0**
Am11e⋆	578.0	134.80	**0.0**	578.0	0.32	**0.0**
Am11f⋆	825.5	134.10	**0.0**	825.5	0.38	**0.0**
Am12a⋆	1493.0	196.40	**0.0**	1493.0	0.38	**0.0**
Am12b⋆	1606.5	195.80	**0.0**	1606.5	0.39	**0.0**
Am12c⋆	2012.5	196.70	**0.0**	2012.5	0.37	**0.0**
Am12d⋆	1107.0	186.00	**0.0**	1107.0	0.42	**0.0**
Am12e⋆	1066.0	191.00	**0.0**	1066.0	0.41	**0.0**
Am12f⋆	997.5	193.90	**0.0**	997.5	0.39	**0.0**
Am13a⋆	2456.5	272.30	**0.0**	2461.5	0.54	0.20
Am13b⋆	2864.0	271.50	**0.0**	2864.0	0.52	**0.0**
Am13c⋆	4136.0	262.40	**0.0**	4137.0	0.53	0.02
Am13d⋆	6164.5	266.50	**0.0**	6164.5	0.51	**0.0**
Am13e⋆	6502.5	267.90	**0.0**	6516.0	0.51	0.21
Am13f⋆	7699.5	265.20	**0.0**	7699.5	0.64	**0.0**
Am14a⋆	2904.0	370.90	**0.0**	2904.0	1.81	**0.0**
Am14b⋆	2736.0	370.40	**0.0**	2736.0	0.83	**0.0**
15⋆	3195.0	498.60	**0.0**	3198.0	0.79	0.09
17	4655.0	870.30	**0.0**	4655.0	1.22	**0.0**
N30-1	4115.0	19829.00	**0.0**	4119.0	12.27	0.10
N30-2	10771.0	11779.50	**0.0**	10801.0	11.02	0.28
N30-3	22697.0	9656.60	0.02	22716.5	12.13	0.11
N30-4	28390.0	9712.90	**0.0**	28485.5	13.07	0.34
N30-5	57393.5	8479.50	**0.0**	57732.0	12.60	0.59
40-1	99543.0	31511.70	0.02	99981.5	45.75	0.46
40-2	300973.5	30784.10	**0.0**	301544.5	38.71	0.19
40-3	416277.0	31291.30	**0.0**	416435.5	35.73	0.04
40-4	207510.0	32492.30	**0.0**	207782.0	48.98	0.13
40-5	193748.0	31121.10	**0.0**	193987.0	51.95	0.12
40-6	1881351.5	30579.80	**0.0**	1882932.5	35.70	0.09
40-7	545239.0	30069.90	**0.0**	549496.0	40.88	0.78
Promedio	101137.1	7456.16	**0.0**	101349.9	**9.78**	0.13

REFERENCIAS

[1] André R. S. Amaral. On the exact solution of a facility layout problem. *European Journal of Operational Research*, 173(2):508 – 518, 2006.

[2] André R. S. Amaral. An Exact Approach to the One-Dimensional Facility Layout Problem. *Operations Research*, 56(4):1026–1033, August 2008.

[3] André R. S. Amaral. The corridor allocation problem. *Computers and Operations Research*, 39(12):3325 – 3330, 2012.

[4] André R. S. Amaral. Optimal solutions for the double row layout problem. *Optimization Letters*, 7(2):407–413, 2013.

[5] André R. S. Amaral. A parallel ordering problem in facilities layout. *Computers & Operations Research*, 40(12):2930–2939, 2013.

[6] André R. S. Amaral. A mixed-integer programming formulation for the double row layout of machines in manufacturing systems. *International Journal of Production Research*, 57:1–14, 04 2018.

[7] André R. S. Amaral. A heuristic approach for the double row layout problem. *Annals of Operations Research*, pages 1–36, 2020.

[8] André R. S. Amaral. A mixed-integer programming formulation of the double row layout problem based on a linear extension of a partial order. *Optimization Letters*, pages 1–17, 11 2020.

[9] Miguel F. Anjos and Anthony Vannelli. Computing globally optimal solutions for single-row layout problems using semidefinite programming and cutting planes. *INFORMS Journal on Computing*, 20(4):611–617, 2008.

[10] Jaewoo Chung and JMA Tanchoco. The double row layout problem. *International Journal of Production Research*, 48(3):709–727, 2010.

[11] T.A. Feo and M.G.C. Resende. A probabilistic heuristic for a computationally difficult set covering problem. *Operations Research Letters*, 8:67–71, 1989.

[12] Sunderesh S Heragu and Andrew Kusiak. Machine layout problem in flexible manufacturing systems. *Operations research*, 36(2):258–268, 1988.

[13] Hasan Hosseini-Nasab, Sepideh Fereidouni, Seyyed Mohammad Taghi Fatemi Ghomi, and Mohammad Bagher Fakhrzad. Classification of facility layout problems: a review study. *The International Journal of Advanced Manufacturing Technology*, 94(1):957–977, 2018.

[14] Philipp Hungerländer and Miguel F. Anjos. A semidefinite optimization-based approach for global optimization of multi-row facility layout. *European Journal of Operational Research*, 245(1):46 – 61, 2015.

[15] Shen Lin and Brian W Kernighan. An effective heuristic algorithm for the traveling-salesman problem. *Operations research*, 21(2):498–516, 1973.

[16] Leonardo D Secchin and André R. S. Amaral. An improved mixed-integer programming model for the double row layout of facilities. *Optimization Letters*, 13(1):193–199, 2019.

[17] Donald M. Simmons. One-dimensional space allocation: An ordering algorithm. *Operations Research*, 17(5):812–826, 1969.

[18] Shengli Wang, Xingquan Zuo, Xueqing Liu, Xinchao Zhao, and Jianqiang Li. Solving dynamic double row layout problem via combining simulated annealing and mathematical programming. *Applied Soft Computing*, 37:303–310, 2015.

Algoritmos de estimación del tiempo de ventana en la recogida de pedidos en almacenes logísticos

Sergio Gil-Borrás*, Eduardo G. Pardo†, Antonio Alonso-Ayuso† and Abraham Duarte† and Ernesto Jiménez-Merino*
*Universidad Politécnica de Madrid
Dpto. Sistemas Informáticos (Madrid, España)
Email: sergio.gil@upm.es y ernesto.jimenez.merino@upm.es
†Universidad Rey Juan Carlos
Dpto. Ciencias de la computación (Madrid, España)
Email: eduardo.pardo@urjc.es, antonio.alonso@urjc.es y abraham.duarte@urjc.es

***Resumen*—En este trabajo se presentan los resultados preliminares del impacto del denominado tiempo de ventana, en el contexto del *Online Order Batching Problem* (OOBP). Este problema consiste en determinar la asignación de pedidos a lotes, en un almacén logístico, para su posterior recogida conjunta, con la característica de que los pedidos llegan dinámicamente al sistema. Este problema ha sido muy estudiado en la literatura de forma general, pero nunca se ha estudiado en profundidad la influencia del tiempo que un operario espera antes de salir a recoger el próximo lote. En este artículo se comparan varios métodos de estimación de tiempo de ventana y se estudia su influencia en algunas de las funciones objetivo más empleadas en la literatura relacionada con el OOBP.**

I. Introducción

En los almacenes logísticos se realizan múltiples tareas, tales como: la recepción de productos, su almacenamiento, la recogida de pedidos, el empaquetado, etc. Entre ellas, la recogida de pedidos supone alrededor de un 55 % del coste total de las actividades que se realizan en un almacén logístico, según los trabajos de varios autores [1, 2]

Este trabajo está centrado en la recogida de pedidos y, más concretamente, en el estudio de un factor poco estudiado hasta la fecha, pero con un alto impacto en la recogida eficiente de los mismos: el tiempo de ventana. Este tiempo, también denominado tiempo de espera, solo aparece en las variantes dinámicas del problema de recogida de pedidos en lotes (OOBP, del inglés *Online Order Batching Problem*) donde no todos los pedidos se conocen de inicio, sino que estos van llegando al almacén con el transcurso del tiempo.

En este contexto, el tiempo de ventana se define como el tiempo que un operario espera desde que está disponible, tras haber finalizado una ruta de recogida y el depósito de mercancías asociado, hasta que inicia una nueva ruta. El sentido de esperar está relacionado con el hecho de que durante la espera pueden llegar nuevos pedidos al sistema, lo que podría permitir realizar una mejor asignación de los pedidos a lotes. Este tiempo de espera puede ser fijo o variable. No obstante, en [3] se concluye que, en general, los algoritmos que utilizan un tiempo de ventana variable, obtienen mejores soluciones que los algoritmos que emplean un tiempo fijo.

En este artículo se comparan diferentes algoritmos para el cálculo del tiempo de ventana variable en el contexto del OOBP. Entre las aportaciones de este trabajo se destacan: i) un estudio del estado del arte con trabajos que incluyen técnicas para determinar el tiempo de ventana; ii) una comparación de diferentes métodos para determinar un tiempo de ventana variable; y iii) la demostración del efecto que tiene el tiempo de ventana sobre dos funciones objetivo ampliamente estudiadas en el contexto del OOBP (la minimización del tiempo que un operario pasa recogiendo pedidos y la minimización del tiempo total empleado para procesar todos los pedidos).

El resto del artículo se organiza de la siguiente forma: en la Sección II se repasan los principales trabajos del estado del arte en los que se proponen o usan algoritmos para determinar el tiempo de ventana. A continuación, en la Sección III, se presenta la descripción del problema y las características del almacén empleado en los experimentos. Después, en la Sección IV se hace un análisis de los algoritmos de tiempo de ventana probados, junto con una descripción del resto de algoritmos necesarios en el contexto del OOBP. En la Sección V se describe cómo se han realizado los experimentos para analizar el comportamiento de este factor. En esta sección también se encuentra una revisión de las instancias usadas. Finalmente, en la Sección VI se encuentran las conclusiones y trabajos futuros.

II. Estado del arte

El uso del tiempo de ventana, en el contexto del OOBP, ha ido evolucionando con el tiempo. Inicialmente nació como una estrategia para la generación de lotes, en la cual, el tiempo que el operario esperaba (ya fuera fijo o variable) determinaba el tamaño del lote. En este contexto, el primer estudio data de 1983 [4] donde se usa la generación de lotes por tiempo de ventana para reducir el tiempo de recogida de pedidos. Posteriormente, en 1991 [5], se presentan varios resultados para su aplicación de manera generalista. En 1997 [6] y en 1999 [7] se ampliaron los trabajos previos, estudiando diversas funciones objetivo, pero aún empleando el tiempo de ventana como estrategia orientada a la generación de lotes. Las funciones objetivo consideradas en estos trabajos fueron:

el tiempo de recogida, el tiempo de servicio y el máximo tiempo que un pedido pasa en el sistema. En ambos artículos, el análisis matemático realizado calcula los límites superior e inferior, la estimación de la varianza y la media de las mismas para cada función objetivo. En 2009 [8] se ampliaron los trabajos publicados en 1997 [6] y en 1999 [7], presentando un modelo basado en teoría de colas, e introduciendo el cálculo del concepto de esperanza del tiempo de espera de un lote (denotado como E[Wj]) para este tipo de problemas. También en 2009 se publica un artículo [9] que proporciona un enfoque analítico, donde se desarrolla el concepto de E[Wj] y se muestra cómo aproximar el tiempo del rendimiento esperado del sistema, para sistemas dinámicos con generación de lotes con tiempo de ventana variable (VTWB, del inglés *Variable Time-Window Batching*) o con la generación de lotes por tiempo de ventana fijo (FTWB, del inglés *Fix Time-Window Batching*). Ya en 2012 [10] se introduce por primera vez un nuevo método para calcular (*a posteriori* y de manera exacta) el momento ideal de salida del operario. Con dicho método, posteriormente, se puede generar un modelo capaz de predecir el mejor momento para la salida del operario a recoger cada lote. En dicho artículo se utilizan procesos de decisión de Markov (MDP, del inglés *Markov Decision Process*) y se comparan con otras heurísticas básicas, en la tarea de minimizar los retrasos en los pedidos y los costes de horas extra de los operarios. También en el año 2012 [11] se presenta por primera vez un algoritmo metaheurístico para la generación de lotes (ILS, del inglés *Iterated Local Search*) junto con una heurística para el cálculo del tiempo de ventana, de manera separada. Este mismo enfoque se ha venido utilizando posteriormente en la literatura, considerando la determinación del tiempo de ventana como un problema en sí mismo. En 2014, en [12], se presentan los resultados de usar la E[Wj] para el cálculo del VTWB, junto con el algoritmo *First Come, First Served* (FCFS) para el proceso de generación de lotes. En 2016 [13] y, posteriormente, en 2017 [14] se presentaron heurísticas híbridas basada en reglas, para diferentes variantes del problema. Estas, unían el cálculo del tiempo de ventana con un algoritmo para la generación de lotes basado en semillas. Finalmente, en el año 2020 [3] se presenta una comparativa sencilla entre métodos de tiempo de ventana fijo y métodos de tiempo de ventana variable.

III. Descripción del problema

Tal y como se ha comentado, el problema estudiado en este trabajo ocurre como una subtarea del OOBP, en el que se aborda el problema de recogida de pedidos en almacenes logísticos, utilizando la política de recogida por lotes y con llegada dinámica de pedidos. Concretamente, el problema consiste en determinar el denominado tiempo de ventana, es decir, el tiempo que debe esperar un operario antes de salir a recoger un lote de pedidos previamente definido, con el objetivo de optimizar una determinada función objetivo.

El tiempo de ventana es un factor poco conocido que influye en la recogida de pedidos. El concepto tiempo de ventana solo aparece en contextos dinámicos, donde los pedidos van llegando continuamente al sistema durante todo el tiempo de procesamiento. En este contexto, inicialmente no se conocen los pedidos que van a llegar al sistema ni los momentos de llegada de cada pedido, sino que estos van llegando a medida que transcurre el tiempo. Los pedidos que ya han llegado al sistema, en cada momento, se agrupan en lotes para su posterior recogida.

El estudio del tiempo de ventana se realiza como parte del proceso de optimización del OOBP, sujeto a su vez a las siguientes restricciones: los pedidos no pueden ser divididos en diferentes lotes; los lotes no pueden ser modificados una vez que el operario salga a recoger el lote correspondiente; la capacidad del lote no puede ser sobrepasada; se considera el caso especial en el que el proceso de recogida de los lotes lo realiza un solo operario en el almacén. Además, es importante destacar, que el problema abordado se estudia en un almacén rectangular de un solo bloque con dos pasillos transversales (uno frontal y otro trasero). Este tipo de almacén ha sido ampliamente utilizado en la literatura del OBP. En la Figura 1 se puede ver un ejemplo de este tipo de diseño de almacén. En ellos, el depósito es el lugar del almacén donde los operarios inician y finalizan la ruta de recogida y se depositan los productos una vez recolectados. En este caso, hay un solo depósito que se encuentra situado en el centro del pasillo trasversal frontal.

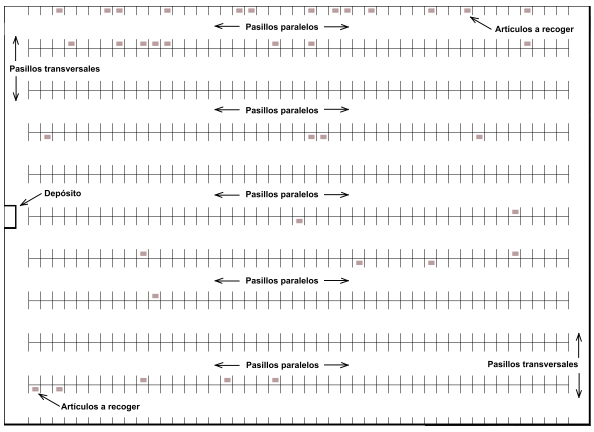


Figura 1. Ejemplo de diseño del almacén estudiado.

Para este trabajo se usan dos funciones objetivo de referencia: minimizar el tiempo de recogida y minimizar el tiempo total. El primero, se calcula como la suma del tiempo que el operario está recogiendo los diferentes lotes. El segundo, además, incluye también el tiempo que los operarios han estado esperando la llegada de nuevos pedidos.

IV. Algoritmos para el problema

Para abordar el problema anteriormente descrito es necesario abordar diferentes tareas: en primer lugar se debe realizar una agrupación de los pedidos disponibles en el sistema en lotes. Una vez formados los lotes de pedidos se tiene que decidir qué lote se recoge primero entre los lotes generados (proceso denominado selección). Una vez que se

ha seleccionado el lote a ser recogido hay que tomar la decisión sobre, si es más conveniente salir a recoger el lote seleccionado, o esperar a que lleguen nuevos pedidos y de este modo mejorar el conjunto de lotes generados (tiempo conocido como tiempo de ventana). Finalmente, una vez que se tiene un lote seleccionado y se ha decidido salir a recogerlo, hay que decidir la ruta que va a seguir el operario por el almacén. A continuación, se repasan los algoritmos utilizados en la comparación experimental. Destacar que, para estudiar las diferentes estrategias de determinación del tiempo de ventana se han tomado como referencia estrategias bien conocidas para el resto de tareas.

IV-A. Generación de lotes

La generación de lotes consiste en agrupar los pedidos en lotes de un tamaño predeterminado máximo, para su posterior recogida. Esta es una de las tareas más estudiadas en la literatura. Para esta tarea se ha seleccionado un algoritmo heurístico muy conocido como es *First Come First Served* (FCFS). Este algoritmo ordena los pedidos según su orden de llegada al sistema. Se empieza a llenar el primer lote con los pedidos más antiguos. Una vez aparece un pedido que ya no cabe en el primer lote, este se da por finalizado y se mete en el siguiente lote. El proceso se repite sin tratar de completar lotes previamente finalizados. Es un algoritmo muy usado en la literatura como punto de referencia.

IV-B. Selección

La tarea de selección consiste en determinar cuál, de entre los lotes disponibles, será el siguiente en recogerse. El método que se emplea en la resolución de este problema es seleccionar el lote que contiene mayor número de artículos, es decir, el que está más lleno. En el caso que haya más de un lote con el máximo número de artículos, se selecciona entre ellos el lote que para su recogida tiene que recorrer una menor distancia de acuerdo con el algoritmo de ruta empleado.

IV-C. Generación de rutas

La tarea de la generación de rutas consiste en determinar una única ruta dentro del almacén que posibilita recoger el conjunto de pedidos de un mismo lote. Hay tres tipos de métodos para generar la ruta de recogida: exactos, heurísticos y metaheurísticos. Esta investigación utiliza un método heurístico muy utilizado en la literatura, fácil de implementar y de entender por parte de los operarios, denominado *S-Shape* (Ver [15, 16] para una descripción detallada del mismo).

IV-D. Cálculo del tiempo de ventana

Por último, se presentan las estrategias estudiadas para el cálculo del tiempo de ventana (objetivo principal de este trabajo). Los algoritmos que calculan el tiempo de ventana pueden clasificarse en algoritmos de tiempo de ventana fijo (FTW, del inglés *Fixed Time-Window*) o algoritmos de tiempo de ventana variable (VTW, del inglés *Variable Time-Window*). Los métodos de tiempo de ventana fijo se caracterizan por establecer un tiempo de espera fijo para todos los operarios y para todas las instancias. Los métodos de tiempo de ventana variable se caracterizan por determinar dinámicamente el tiempo de espera en cada momento. En este trabajo se van es estudiar solamente algoritmos de tipo VTW. Esto es debido a que, en un trabajo previo [3], se demostró que se obtenían mejores resultados que los de tipo FTW. A continuación, se detallan los métodos estudiados:

- Sin Espera: es el método más sencillo. La ruta de recogida de pedidos se inicia tan pronto como esté disponible un operario para la recogida y haya un lote preparado para recogerse.
- Número de pedidos esperando: este método fue empleado por última vez en [3] y consiste en salir a recoger tan pronto como exista un mínimo número de pedidos esperando en el sistema. Se estudian tres variantes de esta estrategia (4, 8 y 16 pedidos esperando).
- Número de lotes generados: este método espera a que se generaren un mínimo número de lotes antes que el operario salga a recogerlos. Se estudian tres variantes de esta estrategia (1, 2, y 3 lotes generados).

V. Experimentación

En esta sección se presentan los experimentos para evaluar los algoritmos de tiempo de ventana anteriormente detallados con el fin de ver el su influencia en dos funciones objetivo ampliamente estudiadas en el contexto del OOBP. Todos los métodos usados en la experimentación, incluidos los del estado del arte, se codificaron en Java 8 y se ejecutaron en una máquina Intel (R) Core (TM) 2 Quad CPU Q6600 2.4 Ghz, con 4 GB de memoria RAM DDR2. El sistema operativo utilizado fue Ubuntu 18.04.1 64 bit LTS.

V-A. Instancias del problema

Para este artículo se han seleccionado un subconjunto de instancias de las propuestas por S. Henn en [17]. Este conjunto de instancias ha sido usado previamente en varios trabajos previos que abordan diferentes variantes del problema [18, 19, 20, 21]. Particularmente, el conjunto de datos reducido consiste en 16 instancias seleccionadas en [3] para la comparación entre métodos de tipo FTW y VTW. El conjunto completo de datos puede ser descargado en http://grafo.etsii.urjc.es/optsicom/oobp/. Las principales características del almacén usado en los experimentos se pueden ver resumidas en la Tabla I. Además de las características del almacén, es también necesario conocer algunos parámetros de trabajo relacionados con el operario. Estos parámetros son recogidos en la Tabla II.

Por último, es necesario determinar el momento en el tiempo para la llegada de los pedidos al sistema. En concreto, los pedidos llegan en un horizonte de tiempo predeterminado siguiendo una distribución exponencial. Esta distribución se deriva en un proceso de Poisson [22]. En concreto, el proceso se define para un horizonte de tiempo de 4 horas (t=4). El número de eventos en que ocurren en un intervalo de tiempo t es una variable aleatoria $X(t)$ con media $E[X(t)] = \lambda * t$. El valor λ es seleccionado dependiendo del número de pedidos

Tabla I
CARACTERÍSTICAS DEL ALMACÉN [17].

	Henn [11]
Política de almacenaje	Aleatorio/ABC
Posición del depósito	Centrado
Tamaño del pedido (#Artículos)	U(5,25)
Peso de los artículos	1
Capacidad del operario (#Artículos)	30/45/60/75
Número de pasillos paralelos	10
Número de artículos por pasillo	2x45
Número total de artículos	900
Longitud de los pasillos paralelos	45m
Ancho de los pasillos	5m

Tabla II
CONFIGURACIÓN DE LOS PARÁMETROS DEL OPERARIO.

Velocidad de viaje del operario	48 m/min
Velocidad de extracción del artículo	6 items/min
Tiempo de preparación del lote	3 min

considerado para cada instancia. En la Tabla III se pueden ver los valores de λ para cada uno de los casos en las instancias.

V-B. Estudio de la minimización del tiempo de recogida

En la Tabla IV se presenta la comparación de la influencia de los diferentes algoritmos de tiempo de ventana en el tiempo de recogida. Concretamente, se reporta: el porcentaje de desviación promedio, el tiempo promedio expresado en segundos y el número de instancias para las cuales el método encuentra el mejor resultado. En este caso, los mejores resultados se obtienen esperando a tener 16 pedidos en el sistema y, el peor resultado, se obtiene con el método "Sin Espera". En estos resultados se observa que según aumenta el tiempo de espera del operario, el tiempo de recogida mejora. Eso se debe a que un mayor número de pedidos en el sistema hace que se puedan formar lotes más compactos, reduciendo así el número de lotes y, por lo tanto, el tiempo de recogida.

Para confirmar la relevancia de los resultados obtenidos, se ha realizado una prueba de rangos de Friedman. El p-valor obtenido de 0,000 indica que las diferencias entre los métodos comparados son significativas. El método de tener 16 Pedidos en cola es el que ocupa la primera posición del ranking. Los resultados del test de Friedman se puede ver en la Tabla V.

Tabla III
VALORES DE λ PARA LA LLEGADA DE LOS PEDIDOS EN 4 HORAS.

	#Pedidos			
	40	**60**	**80**	**100**
4 Horas	0,167	0,250	0,334	0,417

Tabla IV
INFLUENCIA DE LAS ESTRATEGIAS DE TIEMPO DE VENTANA EN LA MINIMIZACIÓN DEL TIEMPO DE RECOGIDA.

	Tiempo de recogida		
	Desviación (%)	**Promedio (s)**	**#Victorias**
16 Pedidos	**0,04 %**	**27057**	**14**
3 Lotes	0,22 %	27090	11
8 Pedidos	0,33 %	27104	10
2 Lotes	0,33 %	27125	8
4 Pedidos	0,96 %	27243	7
1 Lote	0,68 %	27203	6
Sin Espera	4,91 %	27934	1

Tabla V
TEST DE FRIEDMAN PARA EL TIEMPO DE RECOGIDA.

	***Ranking* promedio**
16 Pedidos	**2,69**
3 Lotes	3,25
8 Pedidos	3,50
2 Lotes	3,84
4 Pedidos	3,97
1 Lote	4,16
Sin Espera	6,59

V-C. Estudio de la minimización del tiempo total

En la Tabla VI se puede ver la comparación de la influencia de los diferentes algoritmos de tiempo de ventana respecto al tiempo total. Igual que en la tabla anterior se reportan: el porcentaje de desviación promedio, el tiempo promedio expresado en segundos y el número de veces que cada algoritmo encuentra el mejor resultado. En este caso, la mejor estrategia para minimizar el tiempo total es utilizar la estrategia "Sin Espera". Y el peor método para esta función objetivo es esperar a tener 16 pedidos en el sistema. Como detalle se puede observar que según aumenta el tiempo de espera del operario el tiempo total empeora.

Tabla VI
INFLUENCIA DE LAS ESTRATEGIAS DE TIEMPO DE VENTANA EN LA MINIMIZACIÓN DEL TIEMPO TOTAL.

	Tiempo total		
	Desviacion (%)	**Promedio (s)**	**#Victorias**
Sin Espera	**0,36 %**	**28194**	**9**
4 Pedidos	1,21 %	28430	4
1 Lote	1,34 %	28429	3
2 Lotes	4,30 %	29058	0
8 Pedidos	4,50 %	29189	0
3 Lotes	7,39 %	29717	0
16 Pedidos	10,71 %	30499	0

Nuevamente, para confirmar la relevancia de los resultados obtenidos, se ha realizado una prueba de rangos de Friedman. El p-valor obtenido de 0,000 indica que las diferencias entre los métodos comparados son significativas. El método de "Sin Espera" es el que ocupa la primera posición del ranking. Los resultados del test de Friedman se puede ver en la Tabla VII.

Tabla VII
TEST DE FRIEDMAN PARA EL TIEMPO TOTAL.

	Rango promedio
SinEspera	**1,88**
4 Pedidos	2,38
1 Lote	2,41
2 Lotes	4,28
8 Pedidos	4,63
3 Lotes	5,63
16 Pedidos	6,81

V-D. Comparación entre tiempo de recogida y tiempo total

Como se ha visto anteriormente, el tiempo de recogida y el tiempo total están muy relacionados en entre sí. Siendo una buena forma de estudiar su comportamiento mostrarlos en una misma gráfica, ya que ambas métricas siguen un comportamiento opuesto según qué estrategia de tiempo de ventana se emplee.

En general, se ha observado en los experimentos previos que al aumentar el tiempo de espera del operario el tiempo de recogida se reduce, debido a que se forman lotes más compactos, ya que hay más pedidos disponibles para formarlos. Por el otro lado, se observa que el tiempo total necesario empeora según se aumenta el tiempo de espera. Esta situación, podría también deducirse de dos situaciones límite:

- El tiempo de ventana máximo se obtendría esperando a que lleguen todos los pedidos al sistema antes de salir a recogerlos. En este caso el tiempo de recogida será mínimo ya que al generar todos los lotes conociendo todos los pedidos se produce una solución más compacta en número de lotes. No obstante, esperar hasta el final para empezar la recogida hará que el tiempo total sea muy alto.
- El tiempo de ventana mínimo se obtendría siguiendo una estrategia sin espera. Es decir, el operario sale a recoger los pedidos siempre que haya un lote disponible. En este caso el tiempo de recogida será alto ya que las combinaciones de lotes generadas serán, previsiblemente ,de peor calidad en cuanto a la ruta a seguir. Esto ocurre porque hay muchos menos pedidos en el sistema. En entornos con tasas de llegada bajas puede que los lotes se recojan con un solo pedido disparando así el tiempo de recogida, aunque se minimice el tiempo total.

Para apreciar más claramente la influencia de las estrategias propuestas en las funciones objetivo estudiadas, en la Figura 2 se pueden ver ambas funciones objetivos en una misma gráfica.



Figura 2. Comparación entre las funciones objetivo empleadas (tiempo de recogida y tiempo total).

Como se ve en la gráfica, al comparar el resultado de cada método reportando el valor de ambas funciones objetivo a la misma vez, se obtienen soluciones que, en un enfoque multiobjetivo formarían lo que se conoce como un frente de Pareto [23]. Esto abre una interesante perspectiva para futuras investigaciones, tratando de encontrar nuevos métodos que mejoren ambos objetivos de manera simultánea.

VI. CONCLUSIONES

En este artículo se han comparado diferentes métodos para el cálculo del tiempo de ventana en el contexto del OOBP. En concreto se ha estudiado la influencia de las estrategias propuestas sobre dos funciones objetivo distintas. Cuanto mayor es el tiempo de ventana mejor será el tiempo de recogida, pero peor el tiempo total. Por lo tanto, si se quiere minimizar el tiempo total, será recomendable utilizar estrategias sin espera, o con tiempos de ventana pequeños. Por el contrario, si lo que se desea es minimizar el tiempo que el operario emplea en las rutas de recogida, la mejor estrategia, de entre las estudiadas es esperar a disponer del mayor número de pedidos en el sistema (16 pedidos en este caso).

Los próximos pasos de esta investigación serían ampliar el estudio realizado a un mayor número y diversidad de instancias. Además, se pueden desarrollar nuevos algoritmos que minimicen simultáneamente el tiempo de recogida y el tiempo total en completar la tarea, siendo interesante abordar el problema desde un enfoque multiobjetivo. Otras líneas de trabajo consisten en estudiar la dependencia del tiempo de ventana del resto de tareas que afectan en la recogida de pedidos: generación de lotes, selección y generación de rutas. Por último, también sería interesante analizar el comportamiento del tiempo de ventana cuando el número de operarios varía y cuando se emplean otras funciones objetivo como puede ser el balanceo de carga entre los operarios o el tiempo que un pedido pasa en el sistema.

AGRADECIMIENTOS

Esta investigación ha sido parcialmente financiada por: el Ministerio de Ciencia, Innovación y Universidades de España (Refs. PGC2018-095322-B-C22, RTI2018-094269-B-I00

y TIN2016-80350); por la Comunidad de Madrid y Fondo Europeo para el Desarrollo Regional (Refs. P2018/TCS-4566 y P2018/TCS-4499); y por el Programa Propio de I+D+i de la Universidad Politécnica de Madrid (refs. Programa 466A y MISTI/MIT-UPM, BLOCKSWARM).

Referencias

[1] J. A. Tompkins, J. A. White, Y. A. Bozer, and J. M. A. Tanchoco, *Facilities planning*. John Wiley & Sons, 2010.

[2] J. Drury, "Towards more efficient order picking," *IMM monograph*, vol. 1, 1988.

[3] S. Gil-Borrás, E. G. Pardo, A. Alonso-Ayuso, and A. Duarte, "Fixed versus variable time window warehousing strategies in real time," *Progress in Artificial Intelligence*, pp. 1–10, 2020.

[4] E. B. Quinn, "Simulation of order processing waves," *Material Handling Focus*, vol. 83, 1983.

[5] K. Il-Choe and G. Sharp, "Small parts order picking: design and operation," Atlanta, Georgia, EEUU, 1991.

[6] L. C. Tang and E. P. Chew, "Order picking systems: Batching and storage assignment strategies," *Computers & Industrial Engineering*, vol. 33, no. 3, pp. 817 – 820, 1997, selected Papers from the Proceedings of 1996 ICC&IC.

[7] E. P. Chew and L. C. Tang, "Travel time analysis for general item location assignment in a rectangular warehouse," *European Journal of Operational Research*, vol. 112, no. 3, pp. 582–597, 1999.

[8] M. Yu and R. B. M. De Koster, "The impact of order batching and picking area zoning on order picking system performance," *European Journal of Operational Research*, vol. 198, no. 2, pp. 480–490, 2009.

[9] I. Van Nieuwenhuyse and R. B. M. De Koster, "Evaluating order throughput time in 2-block warehouses with time window batching," *International Journal of Production Economics*, vol. 121, no. 2, pp. 654–664, 2009.

[10] Y. Bukchin, E. Khmelnitsky, and P. Yakuel, "Optimizing a dynamic order-picking process," *European Journal of Operational Research*, vol. 219, no. 2, pp. 335–346, 2012.

[11] S. Henn, "Algorithms for on-line order batching in an order picking warehouse," *Computers and Operations Research*, vol. 39, no. 11, pp. 2549–2563, 2012.

[12] X. Xu, T. Liu, K. Li, and W. Dong, "Evaluating order throughput time with variable time window batching," *International Journal of Production Research*, vol. 52, no. 8, pp. 2232–2242, 2014.

[13] J. Zhang, X. Wang, and K. Huang, "Integrated on-line scheduling of order batching and delivery under b2c e-commerce," *Computers & Industrial Engineering*, vol. 94, pp. 280 – 289, 2016.

[14] J. Zhang, X. Wang, F. T. S. Chan, and J. Ruan, "On-line order batching and sequencing problem with multiple pickers: A hybrid rule-based algorithm," *Applied Mathematical Modelling*, vol. 45, no. Supplement C, pp. 271 – 284, 2017.

[15] C. G. Petersen, "An evaluation of order picking routeing policies," *International Journal of Operations & Production Management*, vol. 17, no. 11, pp. 1098–1111, 1997.

[16] R. De Koster and E. Van Der Poort, "Routing order pickers in a warehouse: a comparison between optimal and heuristic solutions," *IIE Transactions*, vol. 30, no. 5, pp. 469–480, May 1998.

[17] S. Henn and G. Wäscher, "Tabu search heuristics for the order batching problem in manual order picking systems," *European Journal of Operational Research*, vol. 222, no. 3, pp. 484–494, 2012.

[18] S. Gil-Borrás, E. G. Pardo, A. Alonso-Ayuso, and A. Duarte, "Grasp with variable neighborhood descent for the online order batching problem," *Journal of Global Optimization*, vol. 78, no. 2, pp. 295–325, 2020.

[19] B. Menéndez, E. G. Pardo, A. Alonso-Ayuso, E. Molina, and A. Duarte, "Variable neighborhood search strategies for the order batching problem," *Computers & Operations Research*, vol. 78, pp. 500–512, 2017.

[20] B. Menéndez, E. G. Pardo, J. Sánchez-Oro, and A. Duarte, "Parallel variable neighborhood search for the min-max order batching problem," *International Transactions in Operational Research*, vol. 24, no. 3, pp. 635–662, 2017.

[21] R. Pérez-Rodríguez, A. Hernández-Aguirre, and S. Jöns, "A continuous estimation of distribution algorithm for the online order-batching problem," *The International Journal of Advanced Manufacturing Technology*, vol. 79, no. 1, pp. 569–588, 2015.

[22] L. von Bortkiewicz and V. I. Bortkevič, *Das gesetz der kleinen zahlen*. BG Teubner, 1898.

[23] P. Ngatchou, A. Zarei, and A. El-Sharkawi, "Pareto multi objective optimization," in *Proceedings of the 13th International Conference on, Intelligent Systems Application to Power Systems*. IEEE, 2005, pp. 84–91.

XIV Congreso Español de Metaheurísticas, Algoritmos Evolutivos y Bioinspirados

Sesión especial:

Metaheurísticas Aplicadas al Análisis de Redes Sociales



Nuevos algoritmos metaheurísticos para el análisis de la influencia de los usuarios en las redes sociales

Isaac Lozano-Osorio
Universidad Rey Juan Carlos
Madrid, Spain
isaac.lozano@urjc.es

Jesús Sánchez-Oro
Universidad Rey Juan Carlos
Madrid, Spain
jesus.sanchezoro@urjc.es

Abraham Duarte
Universidad Rey Juan Carlos
Madrid, Spain
abraham.duarte@urjc.es

Óscar Cordón
Universidad de Granada
Granada, Spain
ocordon@decsai.ugr.es

***Resumen*—La evolución y la difusión de las redes sociales han atraído el interés de la comunidad científica en los últimos años. En concreto, han surgido nuevos problemas interesantes y difíciles de resolver. En el contexto del marketing viral las empresas y los investigadores tratan de encontrar los elementos que maximizan los beneficios, incrementan visualizaciones, etc. Esta familia de problemas suele conocerse como problemas de maximización de la influencia en redes sociales (SNIM, del inglés *Social Network Influence Maximization*), cuyo objetivo es encontrar los usuarios más influyentes (comúnmente conocidos como semillas) en la red social, simulando un modelo de difusión de influencia. Como se sabe que SNIM es un problema $\mathcal{NP}$-difícil, y la mayoría de las redes a analizar son considerablemente grandes, este problema se ha convertido en un verdadero reto para la comunidad científica. Diferentes trabajos han intentado resolverlo mediante enfoques heurísticos o metaheurísticos, como los algoritmos evolutivos basados en simulaciones del mecanismo de propagación, pero no son capaces de resolver redes de gran tamaño del mundo real debido a sus limitaciones en tiempo de computación. El principal inconveniente de este problema de optimización reside en el esfuerzo computacional necesario para evaluar una solución. Dado que la infección de un nodo se evalúa de forma probabilística, el valor de la función objetivo requiere de una simulación de Monte Carlo para analizar la robustez del método, resultando un proceso computacionalmente complejo. Nuestra propuesta trata de superar estas limitaciones considerando un algoritmo metaheurístico basado en el marco del *Greedy Randomized Adaptive Search Procedure* (GRASP). Esta metodología se divide en dos fases: la fase de construcción, basada en características estáticas de la red para aumentar su eficiencia al no requerir realizar ninguna simulación, y la fase de optimización local, que consiste en una búsqueda local limitada para encontrar a los usuarios más influyentes basándose en movimientos de intercambio. Los experimentos realizados en 6 conocidos conjuntos de datos de redes sociales con 5 tamaños diferentes de conjuntos de semillas confirman que el algoritmo propuesto es capaz de proporcionar resultados competitivos en términos de calidad y tiempo de computación al compararlo con los mejores algoritmos encontrados en el estado del arte.**

***Index Terms*—Sistemas de información, redes sociales, maximización de la influencia, teoría de grafos, marketing viral, GRASP.**

I. Introducción

Hoy en día, las redes sociales (RRSS) son utilizadas diariamente por millones de usuarios, mientras que el número de usuarios sigue creciendo exponencialmente. Este crecimiento se debe a la cantidad de datos generados por los usuarios y, por lo tanto, los problemas clásicos relacionados con las RRSS se están volviendo computacionalmente más difíciles. Una red social puede definirse como la representación de las interacciones sociales que puede utilizarse para estudiar la propagación de ideas, la detección de grupos, la dinámica de los vínculos sociales, la propagación de enfermedades, el marketing viral o la publicidad, entre otros [1, 2, 3, 4, 5]. Formalmente, una red social se modela con un grafo $G(V, E)$ donde el conjunto de nodos V representa a los usuarios y cada relación entre dos usuarios se modela como un par $(u, v) \in E$, con $u, v \in V$ que indica que el usuario u puede transmitir información (o infectar) al usuario v.

Kempe et al. [6] formalizaron originalmente el modelo de influencia para analizar cómo se transmite la información entre los usuarios de una red social. Dada una red con n nodos donde las aristas representan el proceso de difusión o propagación en dicha red, la tarea consiste en elegir un conjunto de semillas S de tamaño $k < n$ con el objetivo de maximizar el número de nodos de la red que son influenciados por el conjunto de semillas S.

Este problema se formuló inicialmente en [7] y posteriormente se demostró que es $\mathcal{NP}$-difícil para la mayoría de los modelos de difusión de influencia (MDI) [8]. Al igual que con muchos otros problemas $\mathcal{NP}$-difíciles, se han considerado algoritmos heurísticos y metaheurísticos, como los algoritmos voraces y evolutivos, para resolver el problema explorando eficazmente el espacio de soluciones, evitando el análisis de cada posible subconjunto de nodos.

La información representa todo aquello que puede ser transmitido entre pares conectados dentro de una red. El nivel de influencia de un nodo viene determinado por el proceso de adopción o propagación. Se pueden encontrar varios MDI en la literatura, pero el más extendido es el Modelo de Cascada Independiente [6], un MDI probabilístico donde la arista (u, v) tiene un peso P que representa la probabilidad con la que u es capaz de influir en v.

Este trabajo presenta una novedosa aproximación metaheurística para abordar este problema, que nos permite encontrar soluciones de alta calidad en el contexto del SNIM en tiempos de computación reducidos. Nuestro principal objetivo es diseñar un algoritmo eficiente en el que se minimice el uso de la simulación Monte Carlo necesaria para la aplicación del MDI, aumentando así la eficiencia del algoritmo.

El resto del trabajo está estructurado como sigue. La sección

2 formaliza el problema y revisa la literatura relacionada. El enfoque propuesto se describe con detalle en la sección 3. En la sección 4 se presentan los resultados experimentales considerando un conjunto de datos público que ha sido utilizado previamente para esta tarea. Los resultados obtenidos se comparan con los algoritmos más populares para SNIM y demuestran claramente la eficacia de la metodología propuesta. Por último, en la sección 5 se extraen algunas conclusiones derivadas de esta investigación.

II. Estado del arte

En esta sección se introducen algunos trabajos relacionados con SNIM y MDI y se realiza un breve estudio de los métodos existentes para resolver este problema, basados en heurísticas o en algoritmos de inteligencia computacional.

Richardson y Domingos [7] formularon inicialmente el problema de selección de nodos objetivo en RRSS. Sin embargo, Kempe et al. [6] intentaron resolver el problema SNIM, formulándolo como un problema de optimización discreto. Se ha demostrado que el cálculo de la propagación de la influencia de un conjunto de nodos dado es $\mathcal{NP}$-difícil [8]. Kempe et al. [6] propusieron un algoritmo voraz llamado *greedy hill-climbing algorithm* con una aproximación de $1-1/e$. Además, también demostraron que este algoritmo voraz puede obtener una solución dentro del 63 % de la optimalidad bajo tres MDIs comúnmente utilizados como el Modelo de Cascada Independiente (MCI), el Modelo de Cascada Ponderada (MCP), y el modelo de Umbral Lineal (UL). El MCI es uno de los MDI más populares y es el que se utiliza en esta contribución. Considera la transmisión de la influencia a través de la red en forma de árbol, donde los nodos semilla son las raíces.

La forma más extendida de evaluar la propagación en MCI es una simulación de Monte Carlo. Sin embargo, incluso una sola iteración de la simulación en MCI consume bastante tiempo. El algoritmo 1 muestra el pseudocódigo de la simulación Monte Carlo para evaluar la propagación de la información a través de la red dado un conjunto de semillas. El algoritmo realiza un número de iteraciones predefinidas IT, encontrando en cada iteración cuáles son los nodos que reciben la información dado un conjunto de semillas S (pasos $2-18$). Inicialmente, el conjunto de nodos $A^\star$ alcanzado por el conjunto semilla inicial, S, es realmente el conjunto semilla (paso 3). A continuación, el método itera hasta que no se infecten nuevos nodos (pasos 5-16). En cada iteración, se recorren los nuevos nodos infectados para comprobar si son capaces de transmitir información a los nodos no infectados (pasos 8-12). Para cada nodo no infectado adyacente a uno de los nuevos nodos infectados, se genera un número aleatorio. Si este número es menor que la probabilidad de infección P, entonces se infecta (pasos 9-11). Al final de la iteración, se actualiza el conjunto de nodos infectados (paso 14) así como los nodos infectados en la iteración anterior (paso 15). Finalmente, el algoritmo devuelve el número medio de nodos infectados durante el número predefinido de iteraciones (paso 19). Este valor, llamado *difusión de la influencia*, se considera como la función objetivo a optimizar por el algoritmo voraz. Es decir, el conjunto de semillas que maximiza el valor de propagación compondría la solución óptima del problema. Obsérvese que, como la infección es un proceso estocástico, el MCI debe ejecutarse varias veces para lograr una estimación adecuada de la media de la variable, lo que da lugar a una simulación de Monte Carlo.

Como consecuencia de coste computacional, Kempe et al [6] también propusieron varias heurísticas voraces basadas en medidas de análisis de redes sociales como la centralidad de grado y de centralidad de cercanía [9]. Cuando la medida considerada es el grado del nodo, el algoritmo se denomina *heurística de alto grado*.

Algorithm 1 $MCI(G=(V,E),S,P,IT)$

```
1:  influencia ← ∅
2:  for i ∈ 1...IT do
3:      A* ← S
4:      A ← S
5:      while A ≠ ∅ do
6:          B ← ∅
7:          for v ∈ A do
8:              for (u, v) ∈ E do
9:                  if rnd(0, 1) ≥ p then
10:                     B ← B ∪ u
11:                 end if
12:             end for
13:         end for
14:         A* ← A* ∪ B
15:         A ← B
16:     end while
17:     influencia ← influencia + |A*|
18: end for
19: return influencia/IT
```

Posteriormente se propusieron varias extensiones de esos primeros algoritmos voraces. En particular, Leskovec et al. [10] introdujeron la selección perezosa rentable (CELF), que explota la propiedad de submodularidad para reducir significativamente el tiempo de ejecución del algoritmo *greedy hill-climbing*, llegando a ser más de 700 veces más rápido que el algoritmo *greedy hill-climbing*. La razón es que la expansión de cada nodo se calcula a priori y sólo es necesario volver a calcularla para unos pocos nodos. Por otra parte, Chen, Wang y Yang [11] utilizaron el concepto de heurística de descuento de grado para optimizar la heurística de alto grado. La función de selección voraz considera la redundancia entre los nodos probablemente influenciados y no cuenta los alcanzados por los nodos semilla ya seleccionados para desarrollar una mejor estimación de la propagación total.

En [12] se propuso un nuevo algoritmo llamado CELF++ con el objetivo de mejorar la eficiencia del CELF original, siendo este un 35-55 % más rápido que CELF. En este trabajo se reimplementó cuidadosamente el algoritmo CELF++ siguiendo las instrucciones de [12] y se probó con diferentes instancias, confirmando los resultados publicados en [13].

Analizando el estado del arte, los enfoques metaheurísticos más complejos suelen dar lugar a mejores soluciones que los enfoques voraces simples. Yang et al. [14] propusieron un algoritmo de optimización de colonias de hormigas basado en un modelo probabilístico parametrizado para abordar el problema SNIM. Utilizaron los métodos de centralidad de grado, centralidad de distancia e influencia simulada para determinar los valores heurísticos.

Mientras tanto, el método basado en temple simulado presentado en [15] aplica dos métodos heurísticos para acelerar el proceso de convergencia, junto con un nuevo método de cálculo de la propagación de la influencia para acelerar el algoritmo propuesto. En [16], se propuso un algoritmo de Optimización Evolutiva Multiobjetivo [17] para SNIM, donde los dos objetivos considerados eran (i) maximizar la influencia de un conjunto de semillas, y (ii) minimizar el número de nodos en el conjunto de semillas.

Como se ha dicho, se han propuesto algunas heurísticas como soluciones para ahorrar tiempo en las decisiones voraces: aleatoria, de grado y de centralidad [6]. La heurística aleatoria selecciona los nodos al azar, sin considerar la influencia de los nodos, para formar el conjunto de semillas en la red. Las heurísticas de grado y centralidad se derivan de la definición de la influencia de los nodos en el análisis de redes sociales [9]. La heurística de centralidad de grado suele producir resultados menos precisos para el problema SNIM. Basada en la heurística de alto grado, otra heurística, se dirige al problema SNIM teniendo en cuenta el conocimiento previo de los vecinos del nodo [11].

Los métodos metaheurísticos mencionados son capaces de obtener buenos resultados pero requieren grandes tiempos computacionales. Nuestra propuesta considera un método equilibrado basado en el MCI. Con el objetivo de reducir el tiempo de computación requerido, consideramos la metodología GRASP, que combina buenas soluciones heurísticas que se generan rápidamente y un método de búsqueda local eficiente que minimiza el número de evaluaciones de Monte Carlo para mejorar la solución inicial.

III. Greedy Randomized Adaptive Search Procedure

Greedy Randomized Adaptive Search Procedure (GRASP) es una metaheurística desarrollada a finales de los años 80 [18] e introducida formalmente en Feo et al. [19]. Remitimos al lector a [20, 21] para un estudio completo de los últimos avances en esta metodología. GRASP es una metaheurística multi-arranque dividida en dos etapas distintas. La primera es una construcción voraz, aleatoria y adaptativa de una solución. La segunda etapa aplica un método de mejora para obtener un óptimo local con respecto a una determinada vecindad a partir de la solución construida. Estas dos fases se repiten hasta que se cumple un criterio de terminación, devolviendo la mejor solución encontrada.

III-A. Fase de construcción

La fase de construcción de GRASP se dedica a generar una solución inicial y suele estar guiada por una función de selección voraz que ayuda al método constructivo a seleccionar los siguientes elementos a incluir en la solución parcial. El algoritmo 2 representa el pseudocódigo del procedimiento constructivo propuesto.

El método parte de una solución vacía S (paso 1), creando la lista de candidatos, LC, con el conjunto completo de vértices V (paso 2). A continuación, el método constructivo añade iterativamente nuevos elementos a la solución hasta que sea factible (pasos 3-11). En cada iteración, se evalúa el valor mínimo y máximo de la heurística voraz (pasos 4-5). A continuación, se calcula un umbral μ (paso 6). Este umbral es necesario para crear la Lista Restringida de Candidatos (LRC) con los nodos más prometedores (paso 7). Finalmente, se selecciona el siguiente nodo al azar de la LRC (paso 8), incluyéndolo en la solución (paso 9) y actualizando la LC (paso 10). El método termina devolviendo la solución construida S (paso 12).

Algorithm 2 $GRASP(G = (V, E), \alpha)$

1: $S \leftarrow \emptyset$
2: $LC \leftarrow$ V
3: **while** $|S| < K$ **do**
4: $\quad g_{min} \leftarrow min_{u \in LC} g(u)$
5: $\quad g_{max} \leftarrow max_{u \in LC} g(u)$
6: $\quad \mu \leftarrow g_{max} - \alpha \cdot (g_{max} - g_{min})$
7: $\quad LRC \leftarrow v \in LC : g(v) \geq \mu$
8: $\quad u \leftarrow rnd(LRC)$
9: $\quad S \leftarrow S \cup \{u\}$
10: $\quad LC \leftarrow LC \setminus \{u\}$
11: **end while**
12: **return** S

La función heurística voraz (usada en los pasos 4-5) es una de las características que mejorarán la solución GRASP. La aproximación que utilizamos en este problema es una heurística basada en los vecinos de primer y segundo grado de un determinado usuario, normalmente conocido como 2 pasos en el análisis de RRSS. Dado un usuario u, este método calcula la suma del grado de salida de u, d_u^+ y el grado de salida de los vecinos descartando los nodos repetidos. En el gráfico representado en la figura 1, el valor heurístico del nodo 1 será $d_1^+ + d_5^+ = 1 + 3 = 4$, el valor heurístico del nodo 5 es 5 debido a $d_1^+ + d_2^+ + d_3^+ + d_4^+ + d_5^+ + d_6^+ = 0 + 1 + 0 + 0 + 4 = 5$. Por lo tanto, el valor de cada nodo se calcula como $[4, 5, 4, 4, 5, 2, 1, 1]$. Luego se elige el nodo con el valor más alto y se actualizan los nodos restantes restando el grado del nodo a sus vecinos (ignorando las aristas con usuarios que han sido seleccionados). En el ejemplo, el usuario seleccionado será el 5, porque es el valor más alto, y los grados de los nodos se recomponen como $[0, 1, 1, 1, X, 1, 1, 1]$. Cabe destacar que el nodo 5 ya está seleccionado y, por tanto, no es necesario calcularlo.

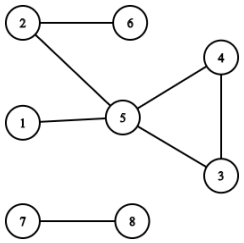


Figura 1. Figura de ejemplo de un grafo con 8 usuarios y 7 relaciones entre ellos para mostrar la función heurística propuesta para evaluar a cada usuario.

III-B. Fase de búsqueda local

La segunda fase de GRASP consiste en mejorar la solución generada por el procedimiento constructivo con el objetivo de alcanzar un óptimo local. Esta fase puede llevarse a cabo utilizando procedimientos de búsqueda local simples o heurísticas más complejas. Debido a la complejidad del problema considerado, proponemos un procedimiento de búsqueda local limitada.

Para definir correctamente una búsqueda local, primero tenemos que introducir la vecindad considerada. La vecindad de una solución puede definirse como el conjunto de nuevas soluciones que pueden alcanzarse realizando un único movimiento sobre la solución original. Por lo tanto, hay que definir un movimiento para presentar la vecindad. En el contexto de este problema, se considera el movimiento de intercambio, que sustituye un nodo seleccionado por otro no seleccionado. Este movimiento de intercambio se define formalmente como:

$$Swap(S, v, u) = S \setminus \{v\} \cup \{u\}$$

Entonces, la vecindad N_s de una solución dada S se define como el conjunto de soluciones que se pueden alcanzar realizando un único movimiento de intercambio a S. Más formalmente,

$$N_s(S) = \{Swap(S, v, u) \forall\ v \in S \wedge \forall\ u \in V \setminus S\}$$

Una vez definida la vecindad, hay que aclarar finalmente en qué orden se explora. En concreto, proponemos recorrer aleatoriamente la vecindad siguiendo un enfoque de *first improvement*, donde se realiza el primer movimiento de mejora.

Debido al gran tamaño del vecindario resultante, $k \times (n-k)$, se presenta una búsqueda local limitada con el objetivo de reducir el número de soluciones exploradas dentro de cada vecindario. El espacio de búsqueda explorado se limita reduciendo el porcentaje de nodos considerados para el movimiento de intercambio.

El número N de soluciones vecinas a explorar para cada solución se define mediante el mínimo entre el tamaño de la vecindad y un parámetro de búsqueda χ, es decir, $\min(|N_s(S)|, \chi)$. Cuanto mayor sea el valor de χ, mejor será la calidad de las soluciones, pero también será mayor el tiempo de cálculo necesario para alcanzarlas. El valor de χ se probó como $\{10, 25, 30\}$.

IV. Experimentos

Esta sección describe el diseño experimental propuesto para validar esta aproximación y muestra los resultados obtenidos. La información relevante sobre las instancias utilizadas se recoge en la tabla I. Todas ellas están disponibles públicamente en Stanford Network Analysis Project (SNAP): https://snap.stanford.edu/. Todos los experimentos se han realizado en un ordenador con un Intel i7-4720HQ (2,50 GHz) y 16GB RAM.

Tabla I
Instancias

Instancia	***Nodos***	***Aristas***	***Diámetro***
p2p-Gnutella31	62586	147892	11
ca-AstroPh	18772	198110	14
ca-CondMat	23133	93497	14
cit-HepPh	34546	421578	12
email-Enron	36692	183831	11
email-EuAll	265214	420045	14

En primer lugar, es importante mostrar los valores utilizados para el algoritmo MCI con Monte Carlo y el número de iteraciones realizadas para obtener el mejor valor con GRASP. En todos los experimentos, GRASP construye 100 soluciones y MCI se ejecuta durante 100 iteraciones (IT) con una probabilidad de infección (P) de 0,01 %. Estos valores de los parámetros son los más extendidos en la literatura. El número de nodos semilla K para conformar una solución se selecciona en el rango $K = \{10, 20, 30, 40, 50\}$, obteniendo así $6 \times 5 = 30$ instancias de problemas diferentes (6 redes y 5 tamaños de conjuntos de semillas).

Los experimentos se dividen en dos partes: la experimentación preliminar y la final. La primera se refiere a los experimentos realizados para seleccionar los mejores parámetros para configurar el algoritmo, mientras que la segunda valida la mejor configuración, comparándola con los mejores métodos encontrados en el estado del arte.

Todos los experimentos reportan las siguientes métricas: Promedio, el valor promedio de la función objetivo; Tiempo (s), el tiempo de cómputo promedio requerido por el algoritmo en segundos; Desv(%), la desviación media con respecto a la mejor solución encontrada en el experimento; y # Mejores, el número de veces que el algoritmo encuentra la mejor solución del experimento.

La experimentación preliminar se ha realizado con un pequeño conjunto de 10 instancias (CA-AstroPh y CA-CondMat con $K = \{10, 20\}$; y Email-Enron y Email-EuAll con $K = \{30, 40, 50\}$) para evitar el sobreajuste. Esta selección abarca un 33,33 % del conjunto global y proporciona suficiente variabilidad en instancias y valores de K.

El propósito del experimento preliminar es establecer el valor de α en la fase constructiva del GRASP, probando los valores $\alpha = \{0.25, 0.5, 0.75, RND\}$, donde RND indica que se elige un valor aleatorio para α en cada construcción. Los resultados obtenidos se muestran en la tabla II. Como se puede observar, el constructivo con el valor de α aleatorio proporciona los mejores resultados tanto en calidad como en

tiempo de ejecución, por lo que es el valor seleccionado para utilizar en la fase de construcción.

Tabla II
RESULTADOS OBTENIDOS EN LA COMPARATIVA PARA ENCONTRAR EL MEJOR VALOR DE α EN LA FASE DE CONSTRUCCIÓN.

α	Promedio	Tiempo (s)	Desv(%)	# Mejores
0.25	406.58	**20.87**	2.12	7
0.50	407.29	23.23	1.87	7
0.75	409.23	28.42	1.02	9
RND	**411.75**	20.92	**0.00**	**10**

La evaluación comparativa entre el mejor diseño GRASP y CELF se reporta en la tabla III. Obsérvese que el tiempo de ejecución es menor para GRASP en todos los casos excepto en K=50 donde es mayor pero a costa de obtener una solución más precisa que CELF. En todas las ejecuciones, la calidad es igual o superior y el tiempo de ejecución es muy competitivo. La última columna muestra el número de instancias donde cada algoritmo, CELF y GRASP, obtiene el mejor resultado.

Tabla III
GRASP VS CELF

Algoritmo	K	Promedio	Tiempo (s)	Desv(%)	# Mejores
CELF	10	243.67	930.53	**0.00**	**6**
GRASP		**243.50**	**49.38**	0.20	5
CELF	20	289.33	890.46	0.06	**5**
GRASP		**289.17**	**252.96**	**0.05**	4
CELF	30	324.50	1209.12	**0.64**	4
GRASP		**325.50**	**570.39**	0.69	**5**
CELF	40	351.33	1409.75	1.65	2
GRASP		**358.00**	**1100.97**	**0.10**	**5**
CELF	50	376.00	**1157.66**	2.01	3
GRASP		**386.83**	1539.02	**0.09**	**5**

La figura 2 ilustra gráficamente que la calidad de GRASP crece más que CELF cuando el tamaño del conjunto de semillas aumenta, pero también podemos ver en la figura 3 cómo el tiempo de ejecución también crece y para $K = 50$ GRASP tiene un tiempo de ejecución mayor que CELF por primera y única vez.

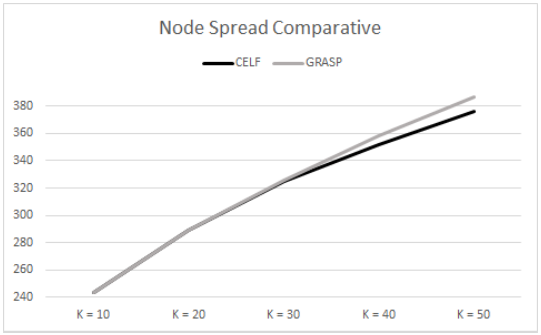


Figura 2. Influencia en CELF vs GRASP

Para confirmar que existen diferencias estadísticamente significativas entre ambos algoritmos, hemos realizado la conocida prueba no paramétrica de Wilcoxon. El valor p resultante,

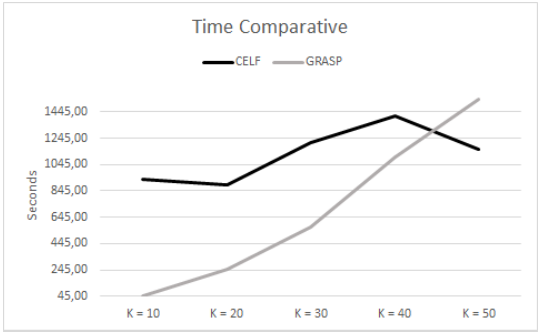


Figura 3. Tiempo en CELF vs GRASP

inferior a 0.001, confirma estas diferencias y, por tanto, la superioridad de la propuesta.

CONCLUSIONES

En este trabajo se ha propuesto un algoritmo GRASP para el problema SNIM. Se ha diseñado una fase de construcción eficiente que reduce el número de simulaciones de Monte Carlo necesarias, así como una búsqueda local limitada para reducir el número de soluciones a explorar y obtener mejores soluciones tanto en términos de calidad como de tiempo de computación. La función heurística proporciona flexibilidad en el diseño del método GRASP para alcanzar un compromiso entre la calidad y el tiempo de computación. GRASP ofrece resultados muy competitivos respecto a CELF, reduciendo drásticamente el tiempo de computación y proporcionando mejores soluciones en promedio.

Por tanto, se puede concluir que es necesario un diseño algorítmico cuidadoso en SNIM para poder tratar con grandes redes sociales sin requerir grandes tiempos de computación. La búsqueda local limitada presentada en este trabajo es capaz de equilibrar la calidad y el tiempo de computación mediante un único parámetro, lo que la hace adecuada para la mayoría de las redes sociales.

AGRADECIMIENTOS

Este trabajo ha sido parcialmente financiado por el "Ministerio de Ciencia, Innovación y Universidades" con la subvención ref. PGC2018-095322-B-C22 y "Comunidad de Madrid" y "Fondos Estructurales" de la Unión Europea con las subvenciones ref. S2018/TCS-4566 y Y2018/EMT-5062.

REFERENCIAS

[1] Stephen F. King y Thomas F. Burgess. «Understanding success and failure in customer relationship management». En: *Industrial Marketing Management* 37.4 (2008), págs. 421-431. DOI: 10.1016/j.indmarman.2007.02.005.

[2] Adam D'angelo et al. *Targeting advertisements in a social network*. US Patent App. 12/195,321. 2009.

[3] Alden S. Klovdahl. «Social networks and the spread of infectious diseases: The AIDS example». En: *Social Science & Medicine* 21.11 (1985), págs. 1203-1216. DOI: 10.1016/0277-9536(85)90269-2.

[4] Sergio Pérez-Peló et al. «On the Analysis of the Influence of the Evaluation Metric in Community Detection over Social Networks». En: *Electronics* 8.1 (2019), pág. 23.

[5] Sergio Pérez-Peló, Jesús Sánchez-Oro y Abraham Duarte. «Finding weaknesses in networks using Greedy Randomized Adaptive Search Procedure and Path Relinking». En: *Expert Systems* (2020), e12540.

[6] David Kempe, Jon Kleinberg y Éva Tardos. «Maximizing the spread of influence through a social network». En: *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '03*. ACM Press, 2003. DOI: 10.1145/956750.956769.

[7] Matthew Richardson, Rakesh Agrawal y Pedro Domingos. «Trust Management for the Semantic Web». En: *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2003, págs. 351-368. DOI: 10.1007/978-3-540-39718-2_23.

[8] David Kempe, Jon Kleinberg y Eva Tardos. «Maximizing the Spread of Influence through a Social Network». En: *Theory of Computing* 11.1 (2015), págs. 105-147. DOI: 10.4086/toc.2015.v011a004.

[9] Stanley Wasserman y Katherine Faust. *Social Network Analysis*. Cambridge University Press, 1994. DOI: 10.1017/cbo9780511815478.

[10] Jure Leskovec et al. «Cost-effective outbreak detection in networks». En: *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '07*. ACM Press, 2007. DOI: 10.1145/1281192.1281239.

[11] Wei Chen, Yajun Wang y Siyu Yang. «Efficient influence maximization in social networks». En: *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '09*. ACM Press, 2009. DOI: 10.1145/1557019.1557047.

[12] Amit Goyal, Wei Lu y Laks V.S. Lakshmanan. «CELF++». En: *Proceedings of the 20th international conference companion on World wide web - WWW '11*. ACM Press, 2011. DOI: 10.1145/1963192.1963217.

[13] Jiaguo Lv et al. «Improved Algorithms OF CELF and CELF++ for Influence Maximization». En: *Journal of Engineering Science and Technology Review* 7.3 (2014), págs. 32-38. DOI: 10.25103/jestr.073.05.

[14] Wan-Shiou Yang y Shi-Xin Weng. «Application of the Ant Colony Optimization Algorithm to Competitive Viral Marketing». En: *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2012, págs. 1-8. DOI: 10.1007/978-3-642-30448-4_1.

[15] Qing-lai Jiang et al. «Preparation of high active Pt/C cathode electrocatalyst for direct methanol fuel cell by citrate-stabilized method». En: *Transactions of Nonferrous Metals Society of China* 21.1 (2011), págs. 127-132. DOI: 10.1016/s1003-6326(11)60688-2.

[16] Doina Bucur et al. «Multi-objective Evolutionary Algorithms for Influence Maximization in Social Networks». En: *Applications of Evolutionary Computation*. Springer International Publishing, 2017, págs. 221-233. DOI: 10.1007/978-3-319-55849-3_15.

[17] Gary B. Lamont Carlos Coello Coello y David A. van Veldhuizen. *Evolutionary Algorithms for Solving Multi-Objective Problems*. Springer US, 2007. DOI: 10.1007/978-0-387-36797-2.

[18] Thomas A Feo y Mauricio G.C Resende. «A probabilistic heuristic for a computationally difficult set covering problem». En: *Operations Research Letters* 8.2 (1989), págs. 67-71. DOI: 10.1016/0167-6377(89)90002-3.

[19] Thomas A. Feo, Mauricio G. C. Resende y Stuart H. Smith. «A Greedy Randomized Adaptive Search Procedure for Maximum Independent Set». En: *Operations Research* 42.5 (1994), págs. 860-878. DOI: 10.1287/opre.42.5.860.

[20] Mauricio G. C. Resende et al. «GRASP and path relinking for the max–min diversity problem». En: *Computers & Operations Research* 37.3 (2010), págs. 498-508. DOI: 10.1016/j.cor.2008.05.011.

[21] Mauricio G. C. Resende y Celso C. Ribeiro. «GRASP: Greedy Randomized Adaptive Search Procedures». En: *Search Methodologies*. Springer US, 2013, págs. 287-312. DOI: 10.1007/978-1-4614-6940-7_11.

Detección de comunidades con un enfoque multi-objetivo utilizando VNS

Sergio Pérez-Peló, Jesús Sánchez-Oro, Antonio González-Pardo, Abraham Duarte
Dept. Computer Sciences
Universidad Rey Juan Carlos
C/Tulipán, S/N,Móstoles, Spain
{sergio.perez.pelo, jesus.sanchezoro, antonio.gpardo, abraham.duarte}@urjc.es
ORCID: 0000-0002-1915-4160, 0000-0003-1702-4941, 0000-0002-1558-9239, 0000-0002-4532-3124

***Resumen*—La detección de comunidades se ha convertido en los últimos años en una de las tareas clave dentro del análisis de redes sociales, ya que ayuda a analizar grupos de usuarios con intereses similares. Esta tarea es útil en diferentes áreas, como la biología (interacciones de genes y proteínas), la psicología (criterios de diagnóstico) o la criminología (detección de fraudes). Este trabajo presenta un enfoque metaheurístico que combina las metaheurísticas *Variable Neighborhood Search* y *Greedy Randomized Adaptive Search Procedure* para la detección de comunidades en redes sociales. En este trabajo, el problema de la detección de comunidades se modela como un problema de optimización bi-objetivo, donde las dos funciones objetivo a optimizar son *Negative Ratio Association* (NRA) y *Ratio Cut* (RC), dos objetivos que se ha demostrado que están en conflicto. Para evaluar la calidad de las soluciones obtenidas, utilizamos la métrica *Normalized Mutual Information* (NMI) para las instancias evaluadas cuya solución óptima es conocida, y la modularidad. Los resultados obtenidos se evalúan mediante métricas muy extendidas en el ámbito de la optimización multi-objetivo, mostrando la propuesta un rendimiento superior que el mejor algoritmo encontrado en la literatura, tanto en calidad como en tiempo de cómputo.**

***Palabras clave:*—Detección de comunidades, VNS, GRASP, Metaheurística, Multi-objetivo**

I. Introducción

En los últimos años, el crecimiento y desarrollo de las redes sociales ha provocado que científicos de diferentes áreas de conocimiento se interesen por el estudio de su estructura y sus implicaciones. Los usuarios de las redes sociales aumentan cada día, lo que las ha convertido en una fuente de datos muy utilizada. Analizando cómo se relacionan los usuarios entre sí y los datos que comparten, podemos obtener información útil para otro tipo de análisis. Por ejemplo, podemos estimar el impacto potencial de una campaña de marketing, cuál es la opinión general sobre un determinado tema, qué piensan los usuarios sobre cierta empresa, persona o servicio, etc.

La mayoría de las redes representan modelos complejos con una gran cantidad de datos e interacciones. El análisis de las redes sociales permite evaluar propiedades como la del mundo pequeño (*small world*) o *free scale*, facilitando la comprensión de aquellas redes pertenecientes al mundo real. Hay una propiedad de las redes sociales que atrae especialmente la mirada de la comunidad científica: la estructura de comunidad [1]. Esta propiedad es capaz de arrojar luz sobre varios problemas de diferentes áreas de investigación, desde las ciencias sociales hasta la biología, entre otras.

Este trabajo propone un algoritmo basado en *Variable Neighborhood Search* (VNS) [2] para resolver el Problema de Detección de Comunidades (CDP) desde una perspectiva multi-objetivo, dando lugar al Problema de Detección de Comunidades Multi-objetivo (MOCDP). Aunque VNS se ha considerado tradicionalmente para la optimización de un solo objetivo, trabajos recientes han adaptado el marco original de VNS para abordar problemas de optimización multi-objetivo, dando lugar a algoritmos robustos que son competitivos con los mejores métodos de la literatura [3].

La mayoría de objetivos utilizados para optimizar la detección de comunidades están en conflicto entre sí, por lo que muchos trabajos previos se han centrado en el enfoque multi-objetivo [4]. Gran parte de los trabajos previos se centran en la adaptación de algoritmos evolutivos conocidos, como NSGA-II, para resolver diferentes problemas de detección de comunidades multi-objetivo. En [5] se propone un algoritmo evolutivo basado en la descomposición para maximizar la densidad interna de las comunidades mientras se minimiza la densidad externa. En [6] se presenta otro algoritmo evolutivo para resolver el MOCDP, considerando como funciones objetivo la maximización de la fuerza intra-enlace de las comunidades y la minimización de la fuerza inter-enlace, que son muy similares a las consideradas en [5]. Obsérvese que, aunque cada trabajo anterior considera diferentes funciones objetivo, son muy similares entre sí, centrándose en localizar en una misma comunidad los nodos más conectados. Hasta donde sabemos, en [7] se presenta el enfoque multi-objetivo más reciente para resolver el MOCDP, considerando las métricas *Negative Ratio Association* y *Ratio Cut* (en la Sección II se definen formalmente estas métricas).

El trabajo se estructura de la siguiente manera: La Sección II define formalmente el problema considerado, así como las métricas propuestas para la evaluación de las soluciones; la Sección III presenta el nuevo procedimiento basado en VNS

Esta investigación está financiada por el "Ministerio de Ciencia, Innovación y Universidades" bajo el proyecto con referencia PGC2018-095322-B-C22, "Comunidad de Madrid" y "Fondos Estructurales" de la Unión Europea bajo los proyectos con referencias S2018/TCS-4566, Y2018/EMT-5062.

propuesto para la detección de comunidades y expone cómo se genera el conjunto de soluciones iniciales con un procedimiento constructivo, así como las estructuras de vecindad consideradas dentro del VNS; la Sección IV introduce los experimentos computacionales realizados para comprobar la calidad de la propuesta; y finalmente la Sección V extrae algunas conclusiones sobre la investigación.

II. Definición del problema

Una red social, conformada por un conjunto de usuarios y las relaciones existentes entre ellos, puede modelarse como un grafo $G = (V, E)$. En este modelo, los usuarios están representados por el conjunto de nodos V, con $|V| = n$, mientras que las relaciones entre dichos usuarios se representan mediante el conjunto de aristas E, con $|E| = m$. Así, una arista $(u, v) \in E$ indica que los usuarios $u, v \in V$ se relacionan en el contexto de la red social. El tipo de relación entre los usuarios depende estrictamente del propósito de la red social (amistad, trabajo, etc.). En este trabajo se consideran relaciones bidireccionales: si existe una relación (u, v), también se contempla la relación (v, u). Es decir, G es un grafo no dirigido.

El objetivo de este trabajo es abordar el Problema de Detección de Comunidades (CDP) siguiendo un enfoque multi-objetivo. Una comunidad $C_i \subseteq V$ dentro de un grafo G se define como un conjunto de usuarios y las relaciones que los conectan. En otras palabras, la comunidad C_i está representada por el subgrafo inducido $G_i = (C_i, E_i)$, donde $E_i = \{(u, v) \in E : u, v \in C_i\}$. El CDP consiste, entonces, en dividir la red social completa en varias comunidades o grupos. Aunque no existe una definición formal de comunidad en la literatura, la definición más aceptada considera que una comunidad es un grupo de usuarios que están estrechamente relacionados entre sí (es decir, comparten algunas propiedades o intereses en la red social).

En términos de grafos, una comunidad con una buena estructura es aquella cuyos nodos están densamente conectados entre sí y escasamente conectados con los nodos que no pertenecen a la comunidad. Dada una comunidad C_i, las aristas que conectan nodos de la misma comunidad se suelen conocer como aristas intracomunitarias, $\mathcal{E}_{\blacktriangleleft}(C_i)$, mientras que las que conectan nodos de diferentes comunidades se denominan aristas intercomunitarias, $\mathcal{E}_{\blacktriangleright}(C_i)$. En términos matemáticos, $\mathcal{E}_{\blacktriangleleft}(C_i) = \{(u, v) \in E : u, v \in C_i\}$ y $\mathcal{E}_{\blacktriangleright}(C_i) = \{(u, v) \in E : u \in C_i \wedge v \notin C_i\}$

Dada una red social, el CDP consiste en asignar a cada usuario a una única comunidad. Cada comunidad se etiqueta con un número entero i, con $1 \leq i \leq c \leq n$, siendo c el número total de comunidades detectadas. Una solución $\mathcal{C}$ para el CDP se modela como el conjunto de comunidades $\mathcal{C} = \{C_1, C_2, \ldots, C_c\}$ de la red. Entonces, una solución para el CDP es factible cuando todos los nodos han sido asignados a una sola comunidad, es decir, $\sum_{i=1}^{c} |C_i| = n$ y $C_i \cap C_j = \emptyset$ para $1 \leq i, j \leq j$ con $i \neq j$. Cabe destacar que el número de comunidades c no es conocido *a priori*, y es tarea del algoritmo encontrar el mejor valor.

La Figura 1(a) muestra un ejemplo de red con 12 nodos y 17 aristas. Las Figuras 1(b) y 1(c) muestran dos soluciones factibles $\mathcal{C}$ y $\mathcal{C}'$, respectivamente, para el CDP, donde cada nodo está señalado con un color diferente, que corresponde a su comunidad (1-verde, 2-rojo, 3-amarillo, 4-azul). La primera solución se representa como $\mathcal{C} = \{C_1, C_2, C_3, C_4\}$, donde $C_1 = \{\mathtt{A}, \mathtt{B}, \mathtt{D}, \mathtt{F}, \mathtt{G}, \mathtt{K}, \mathtt{L}\}, C_2 = \{\mathtt{J}, \mathtt{I}\}, C_3 = \{\mathtt{H}\}, C_4 = \{\mathtt{C}, \mathtt{E}\}$. Del mismo modo, la segunda solución se define como $\mathcal{C}' = \{C'_1, C'_2, C'_3\}$, donde $C'_1 = \{\mathtt{A}, \mathtt{B}, \mathtt{C}, \mathtt{D}, \mathtt{E}\}, C'_2 = \{\mathtt{J}, \mathtt{K}, \mathtt{L}\}, C'_3 = \{\mathtt{F}, \mathtt{G}, \mathtt{H}, \mathtt{I}\}$

Aunque la solución representada en la Figura 1(c) es claramente más llamativa visualmente que la presentada en la Figura 1(b), no existe un criterio común para decidir si una solución presenta una buena detección de comunidades o no. Existen varias métricas ampliamente aceptadas para evaluar la estructura de comunidad de una solución concreta. La Sección IV muestra una descripción en profundidad de las métricas más extendidas.

A diferencia de la mayoría de trabajos, que consideran el CDP como un problema mono-objetivo [8], este trabajo aborda el CDP desde una perspectiva multi-objetivo, dando lugar al Problema de Detección de Comunidades Multi-objetivo (MOCDP). Este enfoque puede ser interesante, ya que podría llevarnos a encontrar nuevas comunidades en las redes más realistas o fiables. En particular, se consideran dos objetivos en conflicto: *Negative Ratio Association* (NRA) y *Ratio Cut* (RC) [7]. El primero mide el porcentaje de aristas intracomunitarias que existe con respecto al tamaño de la comunidad, mientras que el segundo evalúa el porcentaje de aristas intercomunitarias de una comunidad con respecto a su tamaño. En términos matemáticos, el NRA y el RC de un cluster i se definen como:

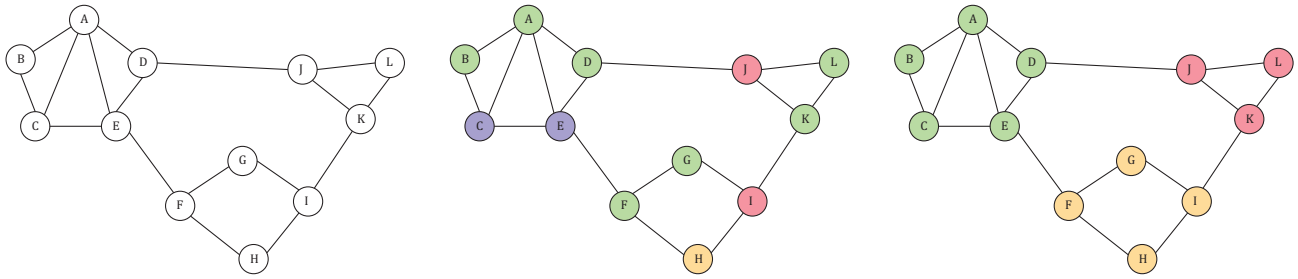
$$NRA(\mathcal{C}) = \sum_{C_i \in \mathcal{C}} -\frac{\mathcal{E}_{\blacktriangleleft}(C_i)}{|C_i|} \qquad RC(\mathcal{C}) = \sum_{C_i \in \mathcal{C}} \frac{\mathcal{E}_{\blacktriangleright}(C_i)}{|C_i|}$$

Siguiendo estas definiciones, una solución con valores pequeños de NRA y RC presenta una buena estructura de comunidad. El MOCDP se centra en minimizar ambos objetivos, NRA y RC, simultáneamente. Analizando las soluciones mostradas en la Figura 1, podemos comprobar que $NRA(\mathcal{C}) = -1.07$ y $RC(\mathcal{C}) = 9.07$ para la solución $\mathcal{C}$, mientras que $NRA(\mathcal{C}') = -3.4$ y $RC(\mathcal{C}') = 1.57$ para la solución $\mathcal{C}'$. De esta forma, la mejor solución con respecto tanto a NRA como a RC es $\mathcal{C}'$, ya que presenta los valores mínimos en ambas funciones objetivo. Por tanto, también podemos concluir que $\mathcal{C}'$ domina a $\mathcal{C}$.

III. Enfoque algorítmico

Este trabajo presenta un algoritmo metaheurístico basado en *Variable Neighborhood Search* (VNS) [2]. La metodología VNS fue diseñada para escapar de óptimos locales realizando cambios sistemáticos de vecindad, aunque no puede garantizar la optimalidad de las soluciones obtenidas.

Originalmente, se ideó para abordar problemas de optimización de un solo objetivo, pero se ha extendido al enfoque



((a)) Red de ejemplo ((b)) Solución $\mathcal{C}$ ((c)) Solución $\mathcal{C}'$

Figura 1: Ejemplo de dos soluciones diferentes para el CDP sobre una red de ejemplo.

multi-objetivo [9]. En este trabajo, adaptamos el VNS multi-objetivo presentado en [3] para resolver el problema de detección de comunidades multi-objetivo, centrándonos en la variante *Basic VNS* (MOBVNS). El algoritmo 1 presenta el marco general de MOBVNS.

Algoritmo 1 $MOBVNS(S, k_{\text{máx}})$

1: $k \leftarrow 1$
2: **while** $k \leq k_{\text{máx}}$ **do**
3: $S' \leftarrow Shake(S, k)$
4: $S'' \leftarrow Improve(S')$
5: $k \leftarrow NeighborhoodChange(S, S'', k)$
6: **end while**
7: **return** S

El algoritmo parte de un conjunto inicial de soluciones no dominadas denotado como S. En el contexto de VNS, el frente inicial puede generarse al azar o utilizando un procedimiento constructivo más elaborado. En nuestro caso, este conjunto inicial se genera mediante un constructivo basado en *Greedy Randomized Adaptive Search Procedure* (GRASP), descrito en la Sección III-A. El segundo parámetro de entrada del algoritmo MOBVNS es la vecindad máxima a explorar durante la búsqueda, $k_{\text{máx}}$.

El algoritmo comienza considerando la vecindad $k = 1$ (paso 1). A continuación, se itera hasta alcanzar el vecindario máximo $k_{\text{máx}}$ (pasos 2-6). En cada iteración, se genera un conjunto perturbado de soluciones S' con el método *Shake* presentado en la Sección III-B. A continuación, se genera S'' como el conjunto de soluciones no dominadas derivadas de la aplicación del método de búsqueda local introducido en la Sección III-C a cada solución incluida en S' para alcanzar un óptimo local de cada solución perturbada. Finalmente, el procedimiento de cambio de vecindad (*Neighborhood Change*) se encarga de seleccionar el siguiente vecindario a explorar.

El método tradicional de cambio de vecindad en VNS mono-objetivo, reinicia la búsqueda desde el primer vecindario ($k = 1$) cada vez que se encuentra una mejora. En caso contrario, la búsqueda continúa en la siguiente vecindad ($k = k+1$). En el contexto de la optimización multi-objetivo, la definición de mejora se modifica ligeramente. En concreto, se considera que se encuentra una mejora si una solución ha podido entrar en el conjunto de las soluciones no dominadas, momento en el que se reinicia la búsqueda desde la primera vecindad.

El algoritmo finaliza cuando no se encuentra ninguna mejora para el conjunto de soluciones no dominadas en ninguno de los vecindarios (es decir, el algoritmo no ha podido introducir una nueva solución en él), devolviendo el conjunto de soluciones no dominadas resultante.

III-A. Generación del conjunto inicial de soluciones no dominadas

El objetivo principal de un buen método constructivo en un problema multi-objetivo es generar un frente con soluciones de alta calidad (es decir, aquellas que no están dominadas) manteniendo la diversidad entre ellas. Con este fin, proponemos un procedimiento constructivo basado en GRASP [10]. Consta de dos fases diferentes: construcción y búsqueda local.

En este trabajo, sólo consideramos la primera etapa de GRASP (es decir, la fase de construcción) para poblar un conjunto inicial de soluciones no dominadas. Este procedimiento comienza creando una comunidad para cada nodo de la red e inicializando el conjunto S de soluciones no dominadas con ella. El siguiente paso corresponde a calcular todas las nuevas comunidades que se pueden crear fusionando dos de las existentes, creando con ellas una lista de candidatos (*CL*). En cada iteración, se construye una lista de candidatos restringida (*RCL*) con los pares de comunidades más prometedores para ser fusionados, siguiendo un cierto criterio voraz y, para aumentar la diversificación, se elige un elemento al azar de la *RCL* para ser incluido en la solución.

Para definir si dos comunidades dadas C_i y C_j deben fusionarse, el algoritmo utiliza una función voraz denominada $g(C_i, C_j)$. Esta función tiene en cuenta el número de aristas que comienzan y terminan en nodos pertenecientes a C_i y C_j, y el tamaño de la comunidad resultante (Ec. 1).

$$g(C_i, C_j) = \frac{|\{(u, v) \in E \ \forall u, v \in C_i \cup C_j\}|}{|C_i \cup C_j|} \qquad (1)$$

Cada solución construida se evalúa para ser considerada en el conjunto de soluciones no dominadas. Cabe mencionar que una solución se considera no dominada si es mejor que otra solución ya incluida en el frente en cualquiera de los dos objetivos.

III-B. Perturbación de las soluciones

El procedimiento de *Shake* se encarga de escapar de óptimos locales en el marco de VNS. Para ello, el método selecciona una solución aleatoria en la vecindad de la solución explorada. Para adaptar adecuadamente este método al contexto de la optimización multi-objetivo, es necesario definir previamente un movimiento básico del MOCDP, que consiste en eliminar un vértice v de su comunidad actual, digamos por ejemplo C_j, e insertarlo en una comunidad diferente, por ejemplo C_i, con $i \neq j$. Más formalmente, dada una solución $\mathcal{C} = \{C_1, C_2, \ldots, C_c\}$, un vértice $v \in C_j$, y una comunidad C_i, con $i \neq j$:

$$Move(\mathcal{C}, v, C_i) = \begin{cases} C_i \leftarrow C_i \cup \{v\} \\ C_j \leftarrow C_j \setminus \{v\} \end{cases} \quad (2)$$

La vecindad de una solución se define como el conjunto de soluciones a las que se puede llegar realizando el mencionado movimiento. En concreto,

$$N(\mathcal{C}) = \{\mathcal{C}' \leftarrow Move(\mathcal{C}, v, C_i) : \forall v \in V \setminus C_i \wedge 1 \leq i \leq c\} \quad (3)$$

Nos basamos en esta definición para introducir la vecindad $N_k(\mathcal{C})$. En particular, $N_k(\mathcal{C})$ se conforma con el conjunto de soluciones que se pueden obtener al realizar exactamente k movimientos básicos consecutivos sobre $\mathcal{C}$.

La solución de salida obtenida en un procedimiento *Shake* no es necesariamente un óptimo local con respecto a la vecindad definida y, por lo tanto, se aplica el método de búsqueda local para optimizar localmente la nueva solución generada.

III-C. Optimización local

En este trabajo se propone un procedimiento de mejora (véase el paso 4 en el Algoritmo 1) que recibe el conjunto perturbado de soluciones, obtenido con el método *Shake*, y devuelve un conjunto donde cada solución es un óptimo local respecto a la vecindad definida.

Para definir una búsqueda local, es necesario comenzar definiendo el movimiento que se va a utilizar. En este trabajo se propone el movimiento $Move(\mathcal{C}, v, C_i)$, definido en la Ecuación 2. Una vez definido el movimiento, es necesario indicar la vecindad que se va a explorar durante el procedimiento de búsqueda local. En este caso, se utiliza la vecindad definida en la Ecuación 3. De esta forma, dada una solución $\mathcal{C}$, la búsqueda local propuesta explorará, en cada iteración, todas las soluciones que se puedan alcanzar aplicando un único movimiento sobre la solución $\mathcal{C}$. Por último, es necesario definir la estrategia con la que se recorrerá la vecindad. En concreto, se propone una estrategia *first improvement* que, en cada iteración, realiza el primer movimiento que lleva a una mejor solución.

Es necesario realizar una adaptación de la búsqueda local tradicional al enfoque multi-objetivo. La búsqueda local propuesta, denominada Independent Search Procedure (ISP), mejora independientemente cada objetivo partiendo de la misma solución. De esta forma, por cada solución evaluada, se aplica un método de mejora para el objetivo NRA y, seguidamente, un método de mejora para el objetivo RC, ambos partiendo de la misma solución inicial. Este enfoque tiene como objetivo poblar el frente de soluciones no dominadas mediante la optimización de ambos objetivos.

IV. Experimentos y resultados

En esta sección se exponen los experimentos realizados para probar la eficacia y eficiencia del algoritmo propuesto, además de compararlo con el mejor método encontrado en la literatura relacionada [7]. Todos los algoritmos se ejecutan sobre dos conjuntos de datos diferentes: redes sintéticas y del mundo real. Para las primeras, hemos utilizado el generador de redes desarrollado por Lancichinetti et al. [11] para construir instancias sintéticas, denominado LFR, donde la distribución de grado de los nodos y el tamaño de la comunidad siguen una ley de potencia altamente configurable. La principal ventaja de estas instancias es que la solución óptima para la estructura de comunidad es conocida en su construcción. Hemos considerado 16 configuraciones diferentes para el generador de redes y hemos generado 2 instancias de red diferentes para cada configuración (generando 32 redes diferentes). En particular, hemos considerado redes con 500 y 1000 nodos, y la probabilidad de generar una arista p se define como $p \in [0.1, 0.8]$ con un intervalo de 0.1.

Para complementar estos casos, consideramos 5 redes del mundo real: *Zacharys karate club* [12] (32 nodos y 78 aristas), *Dolphins social network* [13] (64 nodos y 159 aristas), *American college football* [14] (115 nodos y 613 aristas), *Jazz* [15] (con 198 nodos y 2742 aristas), y *Netscience* [16] [17] (1589 nodos y 2742 aristas). Obsérvese que la solución óptima de las redes de *Zacharys karate club*, *Dolphins social network* y *American college football* se conoce de antemano.

En esta sección se muestran los resultados de la mejor configuración del algoritmo propuesto en comparación con el método del estado del arte. Cabe destacar que esta configuración del algoritmo se ha realizado ajustando los diferentes parámetros de entrada para cada una de las fases mencionadas en la Seción III: α para la construcción de soluciones y k_{max} para la máxima vecindad explorada en VNS. Para una comparativa justa, hemos ejecutado ambos algoritmos con un límite de tiempo de 1800 segundos. Ambos algoritmos se han ejecutado en un ordenador con un núcleo AMD Ryzen 5 3600 AM4 (3.6 GHz) con 16GB de RAM. Todos los algoritmos se han implementado utilizando Java 9.

Este trabajo se enfoca en el MOCDP, en el cual la salida es un conjunto de soluciones no dominadas, en lugar de una única solución. Para comparar el rendimiento de los algoritmos propuestos, utilizamos métricas que evalúan la calidad de los conjuntos de soluciones generados por cada algoritmo. En concreto, hemos considerado cuatro de las métricas multi-objetivo más extendidas [18]. En nuestros experimentos, evaluamos la calidad del conjunto de soluciones no dominadas (S) de un algoritmo específico con respecto a un conjunto de referencia (S'), construido con todas las soluciones no

dominadas encontradas por todos los algoritmos evaluados en el experimento correspondiente. En concreto, las métricas utilizadas son: cobertura, $CV(S, S')$ (maximización), evalúa el número de soluciones dentro del frente de aproximación S que están dominadas por soluciones en S'; hipervolumen, HV (maximización), mide el tamaño del espacio cubierto por el conjunto de soluciones no dominadas; ϵ-indicator, $EPS(S, S')$ (maximización), evalúa la distancia mínima necesaria para transformar cada punto del frente de aproximación bajo evaluación (S) en el punto más cercano del conjunto de referencia S'; y distancia generacional inversa, *IGD+*(S, S') (minimización), mide la distancia del conjunto de soluciones no dominadas (S) al conjunto de referencia obtenido durante el experimento (S'). Además de estas métricas multi-objetivo, en el análisis de redes sociales, existen dos métricas de rendimiento populares que suelen denominarse *Normalized Mutual Information* (NMI) [19] y modularidad (Q) [20].

El NMI requiere un solución óptima conocida, ya que evalúa la diferencia entre la estructura de la comunidad detectada por el algoritmo bajo evaluación y la óptima. Cabe mencionar que la solución óptima es conocida por construcción para todas las instancias de LFR, así como para *karate*, *dolphin* y *football*.

La modularidad puede evaluarse en cualquier red, ya que no depende de una solución óptima conocida. Esta métrica compara la estructura de las comunidades con un grafo aleatorio. Más concretamente, mide la probabilidad de que las comunidades se creen al azar. Por esta razón, la modularidad es especialmente útil para casos del mundo real, como *jazz* o *netscience*, donde la solución óptima es desconocida.

Hay que tener en cuenta que se trata de un problema de optimización multi-objetivo. Por lo tanto, en lugar de tener una única solución, tenemos un conjunto de soluciones no dominadas. Para proporcionar un valor de NMI o Q, seguimos la metodología propuesta en [7]; es decir, recorrer el frente completo encontrando la solución que presenta el mayor valor en cada métrica. Esto implica que la solución que alcanza el mejor valor de Q no es necesariamente la que proporciona el mejor resultado en términos de Normalized Mutual Information (NMI).

Para realizar la comparativa, ejecutaremos ambos algoritmos sobre el conjunto completo de 35 instancias. Primero comparamos los resultados obtenidos con las métricas multi-objetivo previamente mencionadas.

El mejor algoritmo encontrado en el estado del arte, denominado Local information based Multi-Objective Evolutionary Algorithm (LMOEA), es un algoritmo evolutivo multi-objetivo que se basa en la información local de cada nodo. LMOEA requiere de una serie de parámetros establecidos por los autores en el trabajo original [7] considera una población de 100 individuos, 100 generaciones, una probabilidad de cruce de 0.9, una probabilidad de mutación de 0.1 y un tamaño de vecindad de 40 (véase [7] para más detalles). Teniendo en cuenta que estamos comparando algoritmos heurísticos, incluimos un criterio de terminación adicional basado en el tiempo de computación máximo permitido. En concreto, fijamos 1800 segundos como el tiempo máximo empleado en una sola instancia[1]. Si se cumple esta condición de terminación adicional, detenemos el algoritmo correspondiente, devolviendo el conjunto de soluciones no dominadas encontrado durante ese horizonte temporal.

En la Tabla I podemos encontrar los valores medios de las métricas multi-objetivo de *CV*, *HV*, *EPS* y *IGD+*, así como el tiempo medio de cálculo requerido para cada algoritmo evaluado.

Algoritmo	CV	HV	*EPS*	*IGD+*	$T(s)$
MOBVNS	**0.00**	**0.20**	**0.76**	**1018.78**	**214.64**
LMOEA	0.54	0.03	0.40	4281.02	1800.00

Tabla I: Comparativa del frente de referencia obtenido con la mejor configuración de MOBVNS y el LMOEA propuesto por [7]. Los mejores resultados se resaltan en negrita.

Como podemos ver, el MOBVNS propuesto proporciona los mejores resultados en las cuatro métricas. Analizando el tiempo de cálculo, existe una diferencia significativa entre el rendimiento de ambos algoritmos. Mientras que LMOEA emplea el tiempo predefinido de 1800 segundos para cada instancia evaluada, MOBVNS necesita, en promedio, sólo 214.64 segundos, obteniendo soluciones de mayor calidad en tiempos computacionales considerablemente menores.

Una vez comparados ambos métodos utilizando las métricas multi-objetivo clásicas, analizamos además la calidad de su solución considerando el marco experimental descrito en [7]. En particular, primero representamos gráficamente el NMI para cada instancia de LFR.

Para tener conclusiones más robustas, ambos algoritmos fueron ejecutados un total de 20 ejecuciones independientes, reportando los resultados promedio. La Figura 2 muestra estos resultados para $n = 500$ y $n = 1000$, donde p varía de 0.1 a 0.8 con incrementos de 0.1. Como podemos observar en esta figura, MOBVNS presenta soluciones de alta calidad para $p < 0.5$ con valores de NMI cercanos a 1.0. Como era de esperar, para valores mayores de p el comportamiento empeora, ya que estas redes son más difíciles de resolver. LMOEA parece ser más estable en estos casos, oscilando los valores de NMI entre 0.55 y 0.7. De hecho, LMOEA es capaz de superar a MOBVNS en $p = 0.8$ y $n = 1000$.

En el siguiente experimento, comparamos ambos algoritmos sobre el conjunto de instancias del mundo real. Resumimos los resultados sobre el conjunto de instancias en la Tabla II. En concreto, en esta tabla informamos del número de instancias en las que cada algoritmo muestra los mejores resultados utilizando las métricas que acabamos de mencionar, así como el valor medio de estas dos métricas.

Como se puede observar en esta tabla, teniendo en cuenta la modularidad, el algoritmo propuesto (MOBVNS) proporciona la mejor solución en todas las instancias en comparación con LMOEA ejecutado con un límite de tiempo de 1800 segundos.

[1]Se ha considerado este límite de tiempo debido a que en el contexto del CDP la red puede cambiar notablemente si consideramos más de 30 minutos de ejecución.

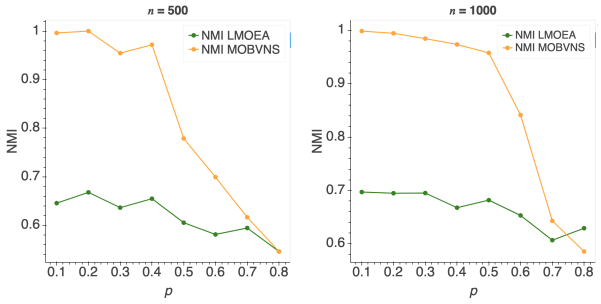


Figura 2: NMI para instancias con n=500 y n=1000

	LMOEA	MOBVNS
# Mejor Modularidad	0	35
# Mejor NMI	10	25
Media Modularidad	0.01	0.32
Media NMI	0.59	0.71

Tabla II: Resumen de los resultados obtenidos por el método propuesto (MOBVNS) y LMOEA cuando se resuelven las 35 instancias consideradas en este trabajo.

En cuanto a NMI, MOBVNS obtiene los mejores resultados en 25 de las 35 redes (72 % de las instancias). Analizando la modularidad media y la NMI media, MOBVNS proporciona de nuevo mejores resultados, ya que estos valores medios son más altos. Estos resultados están respaldados por un p-valor inferior a 0.0001 obtenido en el test no paramétrico por pares de Wilcoxon.

V. Conclusiones y trabajo futuro

En este trabajo, se ha propuesto un nuevo método metaheurístico para la detección de comunidades en redes sociales basado en la hibridación de las metodologías de *Variable Neihborhood Search* (VNS) y *Greedy Randomized Adaptive Search Procedure* (GRASP). El problema se aborda optimizando las métricas de *Ratio Cut* (RC) y *Negative Ratio Association* (NRA) simultáneamente, como un problema de optimización bi-objetivo. La calidad de las soluciones se evalúa considerando las métricas multi-objetivo clásicas, y dos métricas extendidas en el contexto del análisis de redes sociales.

Los experimentos realizados muestran que la combinación de GRASP con VNS en un marco de optimización multi-objetivo es capaz de producir soluciones de alta calidad para el Problema de Detección de Comunidades Multi-objetivo (MOCDP), superando al mejor método encontrado en la literatura, que se basa en un algoritmo evolutivo multi-objetivo (LMOEA, Local Information Multi-Objective Evolutionary Algorithm). Además, la implementación eficiente del algoritmo propuesto es casi diez veces más rápida que el LMOEA original, siendo más adecuado para redes de gran tamaño.

En futuros trabajos, sería interesante analizar el rendimiento de una variante multi-objetivo cuando se compara con los mejores métodos de un solo objetivo encontrados en el estado del arte, incluso incluyendo diferentes objetivos conflictivos. Adicionalmente, el enfoque VNS presentado en este trabajo se probará en nuevas variantes del Problema de Detección de Comunidades, como la Detección Dinámica de Comunidades o la Detección de Comunidades Solapadas, para validar el potencial de este marco de trabajo para abordar los Problemas de Detección de Comunidades.

Referencias

[1] M. Girvan and M. E. Newman, "Community structure in social and biological networks," *Proceedings of the national academy of sciences*, vol. 99, no. 12, pp. 7821–7826, 2002.
[2] P. Hansen and N. Mladenović, "Variable neighborhood search," in *Search methodologies*. Springer, 2014, pp. 313–337.
[3] A. Duarte, J. J. Pantrigo, E. G. Pardo, and N. Mladenovic, "Multi-objective variable neighborhood search: an application to combinatorial optimization problems," *Journal of Global Optimization*, vol. 63, no. 3, pp. 515–536, Nov 2015.
[4] P. Bedi and C. Sharma, "Community detection in social networks," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 6, no. 3, pp. 115–135, 2016.
[5] M. Gong, L. Ma, Q. Zhang, and L. Jiao, "Community detection in networks by using multiobjective evolutionary algorithm with decomposition," *Physica A: Statistical Mechanics and its Applications*, vol. 391, no. 15, pp. 4050–4060, 2012.
[6] C. Shi, Z. Yan, Y. Cai, and B. Wu, "Multi-objective community detection in complex networks," *Applied Soft Computing*, vol. 12, no. 2, pp. 850–859, 2012.
[7] F. Cheng, T. Cui, Y. Su, Y. Niu, and X. Zhang, "A local information based multi-objective evolutionary algorithm for community detection in complex networks," *Applied Soft Computing*, vol. 69, pp. 357 – 367, 2018.
[8] S. Pérez-Peló, J. Sánchez-Oro, R. Martín-Santamaría, and A. Duarte, "On the analysis of the influence of the evaluation metric in community detection over social networks," *Electronics*, vol. 8, no. 1, 2018.
[9] X. Song, D. Jones, N. Asgari, and T. Pigden, "Multi-objective vehicle routing and loading with time window constraints: a real-life application," *Annals of Operations Research*, pp. 1–27, 2019.
[10] T. A. Feo, M. G. C. Resende, and S. H. Smith, "A Greedy Randomized Adaptive Search Procedure for Maximum Independent Set." *Operations Research*, vol. 42, no. 5, pp. 860–878, 1994.
[11] A. Lancichinetti, S. Fortunato, and F. Radicchi, "Benchmark graphs for testing community detection algorithms," *Phys. Rev. E*, vol. 78, p. 046110, 2008.
[12] W. W. Zachary, "An information flow model for conflict and fission in small groups," *Journal of anthropological research*, vol. 33, no. 4, pp. 452–473, 1977.
[13] S. Gregory, "Finding overlapping communities in networks by label propagation," *New journal of Physics*, vol. 12, no. 10, p. 103018, 2010.
[14] M. E. J. Newman, "Modularity and community structure in networks," *Proceedings of the National Academy of Sciences*, vol. 103, no. 23, pp. 8577–8582, 2006.
[15] P. M. Gleiser and L. Danon, "Community structure in jazz," *Advances in complex systems*, vol. 6, no. 04, pp. 565–573, 2003.
[16] R. A. Rossi and N. K. Ahmed, "The network data repository with interactive graph analytics and visualization," in *AAAI*, 2015. [Online]. Available: http://networkrepository.com
[17] M. E. Newman, "Finding community structure in networks using the eigenvectors of matrices," *Physical review E*, vol. 74, no. 3, p. 036104, 2006.
[18] M. Li and X. Yao, "Quality evaluation of solution sets in multiobjective optimisation: A survey," *ACM Computing Surveys (CSUR)*, vol. 52, no. 2, p. 26, 2019.
[19] L. Danon, A. Díaz-Guilera, J. Duch, and A. Arenas, "Comparing community structure identification," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2005, no. 09, pp. P09 008–P09 008, 2005.
[20] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Phys. Rev. E*, vol. 69, p. 026113, Feb 2004.

Análisis de Redes Sociales basado en las Conquistas de César Borgia

1st Javier Fumanal-Idocin
Estadística, Informática, Matemáticas
Universidad Pública de Navarra
Pamplona, España
javier.fumanal@unavarra.es

2nd Oscar Cordón
Andalucian Research Institute
University of Granada
Granada, España
ocordon@decsai.ugr.es

3rd Amparo Alonso-Betanzos
CITIC
University of A Coruña
A Coruña, Spain
amparo.alonso.betanzos@udc.es

4th Humberto Bustince
Estadística, Informática, Matemáticas
Universidad Pública de Navarra
Pamplona, España
bustince@unavarra.es

5th Javier Fernández
Estadística, Informática, Matemáticas
Universidad Pública de Navarra
Pamplona, España
bustince@unavarra.es

***Resumen*—En este trabajo presentamos el modelado de redes sociales y detección de comunidades utilizando como base un evento histórico real, las conquistas de César Borgia en el siglo XV. Para ello, proponemos un nuevo conjunto de funciones, llamadas funciones de afinidad, diseñadas para capturar la naturaleza de las interacciones locales entre cada par de actores en una red. Utilizando estas funciones, desarrollamos un nuevo algoritmo de detección de comunidades, el Borgia Clustering, donde las comunidades surgen naturalmente de un proceso de simulación de interacción de múltiples agentes en la red. También discutimos los efectos del tamaño y la escala de cada comunidad, y cómo pueden ser tomadas en cuenta en el proceso de simulación. Finalmente, comparamos nuestra detección de comunidades con otros algoritmos representativos, encontrando resultados favorables a nuestra propuesta.**

I. Introducción

El análisis de redes se ha convertido en una herramienta importante para estudiar sistemas compuestos por agentes que interactúan, como proteínas o sociedades humanas [1]–[4]. Hay muchos problemas relacionados con las redes sociales, como la difusión de información [5], la detección de círculos sociales [6], la formación de coaliciones [7] o los sistemas de recomendación [8]. Las redes sociales también pueden verse como sistemas multiagente, y es posible estudiar sus propiedades emergentes [9]. Un paso clave para analizar una red es identificar su estructura de comunidad: los grupos de nodos que pueden identificarse como una subpartición funcional del grafo [10], [11]: por ejemplo, un grupo de amigos o un complejo de proteínas [12]. Las comunidades son importantes porque podemos inferir un conocimiento significativo de un nodo o un conjunto de nodos si sabemos si comparten o no la misma comunidad y qué tipo de comunidad es. Existen muchos métodos para realizar la detección de comunidades. Por ejemplo, la propuesta en [13] usa el ratio de conexión internas y externas de grupos de nodos para obtener una rápida división en grupos de la red, o también, los autores de [14] proponen un enfoque de optimización de modularidad para trabajar en redes grandes, siendo este uno de los métodos de detección de comunidades más utilizados.

En este artículo, nuestro objetivo es resolver algunos de los problemas de análisis de redes sociales que hemos identificado tanto en la interacción entre actores como en la detección de la comunidades. Primero, los algoritmos de detección de comunidades actuales no tienen en cuenta ni la diferente naturaleza de los seres humanos involucrados en estas, ni el impacto de esta naturaleza en la estructura de la red. Tampoco tienen en cuenta cómo la escala en redes sociales puede alterar su dinámica. Además, cuando se trata de detección de comunidades, muchos de los algoritmos existentes tienen problemas en grafos densamente conectados y los algoritmos apropiados a menudo consumen mucho tiempo [15].

Para resolver estos problemas, estudiamos cómo los algoritmos inspirados en humanos pueden conducir a una mejor comprensión de la estructura de una red social y sus comunidades. En contraste con la literatura existente, que utiliza una matriz de adyacencia para modelar una red [16], proponemos un conjunto de funciones con el fin de capturar mejor las relaciones entre cada par de actores en el contexto social de la red. Utilizando como base estas funciones de afinidad y el algoritmo gravitacional de clustering [17], hemos desarrollado un nuevo algoritmo para realizar la detección de comunidades, el Borgia Clustering, que debe su nombre a que su comportamiento busca emular las mismas condiciones que se dieron durante las conquistas de César Borgia.

Usando las funciones de afinidad y el Borgia Clustering, nuestro objetivo es tener una mejor comprensión de las interacciones locales entre y cómo pueden afectar la dinámica global y formación de comunidades. De esta manera, podremos generar un dendrograma que refleje fielmente la evolución de la red en el proceso de agrupamiento, y elegir la configuración correcta dentro de él.

El resto del trabajo es la siguiente. En la Sección II explicamos los conceptos básicos del algoritmo gravitacional y la teoría de grafos. En la Sección III explicamos las nuevas

funciones de afinidad y damos un caso práctico de su uso en *República* de Platón. En la Sección IV presentamos el algoritmo Borgia Clustering y el momento histórico que lo inspiró, y en la Sección V comparamos la calidad de nuestra solución con otros algoritmos representativos. Finalmente, en la Sección VI resumimos todo el trabajo y establecemos algunas pautas futuras.

II. Preliminaries

En esta sección explicaremos brevemente algunos de los conceptos ya existentes relacionados con algunas de las nuevas propuestas de este trabajo:

- Algoritmo de clústering gravitacional.
- Teoría de grafos.

II-A. Algoritmo de clústering gravitacional

El algoritmo de clústering gravitacional [17] emplea la ley gravitacional de Newton para modelar la atracción entre distintas observaciones de datos. En este algoritmo, cada observación es una partícula que atrae a las demás según sus distancias y masas. Cuando dos partículas están más cerca que la distancia de colisión, se fusionan en una sola. Su masa es la suma de esas partículas y su posición es su centro de masas. Este proceso se repite hasta que solo existe una partícula. Este algoritmo da como resultado un dendrograma que contiene cada fusión de partículas. Finalmente, la configuración más estable se toma como la resultante. El esquema es el siguiente:

Supongamos que tenemos n particulas $p_1, \ldots, p_n$, con posiciones $s_1, \ldots, s_n \in R^n$. También tenemos dos parámetros: ϵ, que establece la distancia de colisión para dos partículas, y δ que determina el movimiento de la partícula más rápida en cada iteración:

1. Inicialmente asignamos una masa de $m_i = 1$ a cada partícula p_i.
2. Decidimos el valor de ϵ y δ
   - Usamos δ para determinar la duración del delta del tiempo, dt, que significa la velocidad a la que la partícula más rápida se mueve.
   - Si en cualquier momento dos partículas se encuentran a una distancia menor de ϵ se unen en una partícula. La masa de la nueva partícula es la suma de las dos, y su posición el centro de masas.
3. El tiempo inicial se pone a $t = 0$.
4. Repetimos los siguientes pasos (i)-(iv) hasta que sólo quede una partícula.
   (i) En cada intervalo de tiempo $[t, t+dt]$, para cada p_i calculamos la fuerza:

$$g(i,t,dt) = \frac{1}{2} G \sum_{j \neq i} \frac{m_i(t) m_j(t)}{m_i(t)} \frac{s_j(t) - s_i(t)}{|s_j(t) - s_i(t)|^3} dt^2 \quad (1)$$

   donde G es una constante positiva.
   (ii) Por cada partícula i, su nueva posición es:

$$s_i(t + dt) = s_i(t) + g(i,t,dt)$$

   (iii) Incrementamos t a $t + dt$.
   (iv) Si dos partículas i y j están más cerca que ϵ, se fusionan en una, siguiendo el procedimiento anteriormente explicado.

Cuando el algoritmo termine, quedará sólo una partícula. La duración del proceso entero es T. Medimos la duración de cada iteración también, con el objetivo de quedarnos con la configuración que más tiempo aguantó, que es el resultado final.

II-B. Teoría de Grafos

Un grafo G se representa como $G(V, E)$ donde V es un conjunto de vértices y E es un conjunto de aristas que conectan algunos pares de vértices en el grafo G. Hay diferentes tipos de grafos, dependiendo de la información relacionada con cada arista. En caso de que tengamos alguna información sobre la fuerza de las relaciones, llamamos al grafo un grafo ponderado. Si no tenemos tal información, se llama grafo no ponderado. Las aristas también pueden ser no dirigidas, cuando la arista entre $V_i \leftrightarrow V_j$ representa una relación bidireccional, o dirigida, cuando la arista entre $V_i \rightarrow V_j$ puede ser diferente de la arista $V_j \rightarrow V_i$. Los grafos se pueden caracterizar mediante el uso de muchas estadísticas, como el número promedio de conexiones por nodo, las longitudes de ruta promedio entre nodos, etc.

Los grafos se pueden modelar utilizando diferentes representaciones, las más comunes son la matriz de adyacencia o conectividad y la lista de adyacencia. La matriz de adyacencia A de un grafo G es una matriz $N \times N$ donde N es el número de nodos en G. Cada entrada A_{ij} en esta matriz corresponde al valor asociado con la arista $V_i \rightarrow V_j$. Si el grafo no está ponderado, esos valores serán 0 o 1, mientras que si el grafo está ponderado, esos valores serán los pesos correspondientes para cada arista. La lista de adyacencia es similar a la matriz de adyacencia, pero en lugar de almacenar $N \times N$ elementos, almacenamos una lista para cada arista que contiene el resto de nodos que se comunican con ella.

III. Funciones de afinidad para caracterizar relaciones entre actores

Definimos las funciones de afinidad como funciones para dos actores, Actor_x y Actor_y, que establecen una relación mutua usando C:

$$A_C : (Actor_x, Actor_y) \rightarrow [0, 1]$$

Generalmente, esta C es la matriz de adyacencia que cuantifica las relaciones en cada par de actores, aunque C puede ser, por ejemplo, otra matriz construida con funciones de afinidad, o una lista de ellas. La afinidad entre dos actores muestra cuán fuertemente están conectados estos dos. Dado que las afinidades no son necesariamente simétricas, la fuerza de esta interacción depende de quiénes son el emisor y el receptor, como sucede en la interacción humana, por ejemplo amor no correspondido.

Procedemos a enumerar algunas funciones de afinidad:

- **Afinidad de mejor amigo**:la afinidad del actor $Actor_x$ sobre $Actor_y$ se define como el porcentaje de la conectividad total de Y que corresponde a $C_{x,y}$.

$$A_C^{BF}(x,y) = \frac{C_{x,y}}{\sum_{a=1}^{N} C_{x,a}}$$

- **Afinidad de amigo común**: la afinidad entre dos actores se define como la mayor afinidad común a ambos.

$$A_C^{BCF}(x,y) = \frac{Max\{Min(C_{x,z}, C_{y,z})\}}{\sum_{a=1}^{N} C_{x,a}}$$

- **Afinidad de Mejores Amigos para Siempre**: la afinidad de dos actores refleja la durabilidad de la relación en el tiempo.

$$A_C^{FF} = \frac{Max\{Min(C_{x,z}, C_{y,z})\}}{\sum_{a=1}^{N} C_{x,a}}$$

- **Machiavelli Affinity**: la afinidad entre dos actores se basa en la estructura social que se construye alrededor de ellos dos.

$$A_C^{MAC}(x,y) = A_C(x,y) = 1 - \frac{abs(I_x - I_y)}{Max(I_x, I_y)},$$
$$I_a = Sum(Degree(x'))\forall a, C_{a,x'} > 0$$

Existen, principalmente, dos tipos de funciones de afinidad: afinidades personales y afinidades estructurales. Las afinidades personales establecen la fuerza de una conexión interpersonal $Actor_x \rightarrow Actor_y$ usando sus respectivas conexiones y amigos compartidos. Las afinidades estructurales cuantifican la relación de un par de actores en función de las propiedades de sus nodos, como su grado o intermediación. Todas las afinidades aquí propuestas, a excepción de la de Maquiavelo, son afinidades personales.

III-A. Efecto de diferentes funciones de afinidades en República de Platón

Como caso de estudio, consideramos la red de asociación de palabras en un clásico de la literatura: *República*, de Platón. Hemos visualizado la red resultante para cada afinidad diferente (Figura 1). En estas redes, cada nodo corresponde a una palabra diferente en la obra original, y su tamaño es directamente proporcional a su grado. En la red de co-ocurrencias, cada arista representa la cantidad de veces que dos palabras aparecen juntas en un párrafo y en los grafos de afinidad, cada arista es el valor de afinidad para cada par de nodos. Por motivos de claridad, solo las 130 palabras más frecuentes están presentes en cada red.

Dependiendo de la función de afinidad utilizada, las aristas resultantes y sus pesos pueden ser muy diferentes. Como ejemplo, en la Tabla I hemos calculado las cinco afinidades para la entidad "Man" (Por conveniencia, en este trabajo hemos considerado la traducción inglesa y respetamos el nombre original de las entidades procesadas).

Outgoing	Adj.	BF	BCF	FF	MA
Top 1	Justice	Justice	Justice	Life	Life
Top 2	Life	Life	Injustice	Other	Justice
Top 3	Injustice	Injustice	Soul	One	State
Top 4	Soul	Soul	Life	Soul	Men
Top 5	State	State	State	Evil	Soul
Incoming					
Top 1	Justice	Master	Tyrant	Reward	Soul
Top 2	Life	Desire	Desire	God	Justice
Top 3	Injustice	Action	Spirit	Gold	Life
Top 4	Soul	Word	Journey	Work	State
Top 5	State	Case	Master	Protect	Men

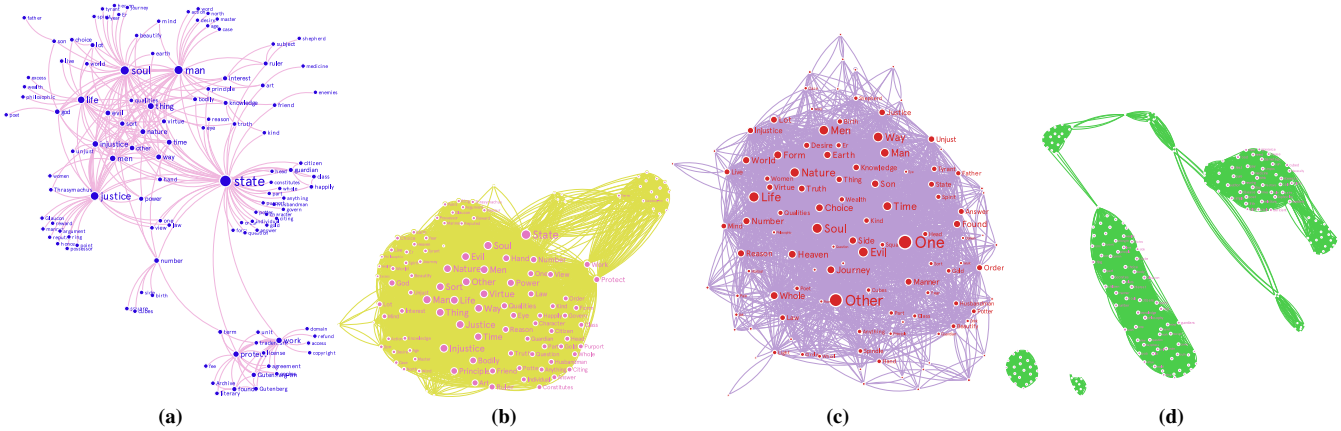
Cuadro I: El efecto de diferentes cálculos de afinidad para el actor "Man" en *República* de Platón. Cada columna muestra los valores más altos de la matriz de adyacencia y diferentes funciones de afinidad para cada arista del actor "Man".

IV. Borgia Clustering

En 1497, César Borgia como Comandante en Jefe del Ejército Papal y al mando de su padre, Alejandro VI, marchó por el centro de Italia, conquistando todos los territorios tradicionalmente vinculados a los Estados Pontificios [19]. Este acontecimiento crucial en la historia del Renacimiento nos dejó no solo momentos de memorable valentía individual y genio táctico, sino también un excelente ejemplo de interacción humana tanto a nivel personal como comunitario.

Nuestro objetivo es generar un algoritmo que parta de una red social cualquiera y simule las mismas dinámicas que generó César Borgia al iniciar su campaña. Concretamente, nos hemos fijados en estas condiciones:

- Todos los países tienen como objetivo crecer. Aunque pueda parecer una cosa trivial, no siempre es así. A veces, los estados prefieren crear y mantener un equilibrio en lugar de romperlo a su favor.
- Algunas partes de Italia son culturalmente más similares a países diferentes. Nápoles, por ejemplo, está mucho más influenciada por la cultura española que Milán, y las futuras uniones dinásticas y conquistas harán evidentes estas diferencias.
- Las diferencias de tamaño y poder hacen que algunas alianzas sean más valiosas que otras. Las repúblicas italianas buscan alianzas no solo en su península sino en tierras de naciones más poderosas. Este comportamiento favorece la creación de bandos enfrentados liderados por grandes potencias, como fue el caso de las numerosas guerras franco-españolas ocurridas en los siglos XV y XVI, lo que puede entrar en conflicto con las afinidades culturales de los distintos países.
- Hubo muchos países de diferente tamaño e importancia involucrados en ese evento histórico. Francia y España fueron los más grandes, seguidos de Milán, Venecia, Florencia y el Reino de Nápoles. También estaban los Estados Pontificios y los "señores independientes" en el



(a) (b) (c) (d)

Figura 1: Distintas formas de representación en red de *República* de Platón. a. Red de coocurencias/Afinidad de mejor amigo. **b**. Red de afinidad de mejor amigo común. **c**. Red de afinidad de mejores amigos para siempre. **d**. Red de afinidad de Maquiavelo. Las redes han sido dibujadas con el algoritmo Force Atlas 2 [18]

centro de Italia. Contrariamente al pensamiento ingenuo, los condados más pequeños no fueron conquistados por los países más grandes, que son los que tienen ejércitos más grandes, sino por los Estados Pontificios.

El algoritmo se basa en el clásico algoritmo *Gravitational Clustering* [17]: cada actor comienza como una comunidad diferente que se acerca a los demás debido al efecto de una fuerza de atracción. Esta fuerza de atracción hará que cada comunidad quiera crecer siempre, de acuerdo a sus afinidades naturales y al valor de cada posible "alianza". A continuación, detallamos todas las modificaciones hechas al algoritmo original.

IV-A. Modificación en las partículas

En una red social, cada partícula no es un punto en el espacio, sino un actor. Los actores tienen un conjunto de propiedades adicionales, generalmente relacionadas con la información semántica disponible para cada uno. La más importante es la conectividad, que nos permite calcular la afinidad de mejor amigo y de mejor amigo común.

Para representar a cada actor, utilizamos una combinación de la matriz de estas afinidades, y una matriz de influencia, S, basada en la primera. Al usar la afinidad de mejor amigo, favorecemos interacciones fuertes por pares, y con la afinidad de mejor amigo común, también favorecemos la formación de comunidades cuyos miembros comparten un gran número de amigos.

Los actores tienen un valor social inicial, m, que inicialmente corresponde a su grado, que es similar al concepto de masa de partículas en el algoritmo original y que refleja su popularidad. Esto nos permite mostrar dos comportamientos sociales bien conocidos: (1) las personas populares son más atractivas socialmente que las personas con menos amigos. Este hecho es llamado ventaja acumulativa en la teoría de redes [20] (desde el punto de vista físico de, una mayor masa implica una mayor fuerza de atracción). (2) Las personas con pocos amigos intentan pasar el rato con gente popular, pero no al revés (una mayor masa también implica menos movimiento).

El concepto de posición es sustituido por el concepto de influencia, denotado como la matriz S, que es construido usando una matriz de afinidad. El valor s_{ij} refleja la influencia que el actor j tiene sobre el actor i. Las autoinfluencias (s_{ii}) se establecen en 1. Los actores interactuarán entre sí a lo largo del tiempo y se acercarán a medida que pase el tiempo debido a esta interacción. Además, las autoinfluencias disminuirán durante la ejecución del algoritmo.

La matriz A_C es inicialmente la misma que la matriz S, pero A_C solo cambia sus valores cuando dos actores chocan y forman una nueva comunidad.

IV-B. Modificaciones en la fusión

La condición para comprobar si dos actores han chocado usando la distancia euclidiana no es buena en este contexto debido a la maldición de la dimensionalidad. Por tanto, hemos utilizado la idea de influencia en lugar de posición para comprobar si dos actores deberían fusionarse o no.

Cada actor comienza siendo completamente influyente sobre sí mismo. Durante la ejecución del algoritmo, los actores se atraen entre sí, provocando una reducción en su propia autoinfluencia y aumentando la influencia que otros actores tienen sobre ellos. Cuando la influencia de un actor sobre otro es mayor que la autoinfluencia de ese actor, esos actores colapsarán en una nueva comunidad.

La fusión de dos actores a, b da como resultado la aparición de un nuevo actor cuyo valor social es la suma de m_a y m_b, cuya posición es su centro de masas de s_a y s_b, y cuyas afinidades son el centro de masas de las afinidades de a y b.

IV-C. Nueva fórmula de atracción

Hemos modificado la fórmula de atracción para tener en cuenta no solo la masa y la distancia, sino todo el conjunto de características presentes en cada actor.

Tenemos en cuenta las relaciones bilaterales utilizando las matrices A_C y S. La matriz A_C devuelve la afinidad por un par de actores/comunidades, y usamos S para calcular las distancias euclidianas entre ellos, de modo que las afinidades más comunes o similares entre dos actores sean, mayor será la fuerza de atracción.

También debemos tener en cuenta el tamaño del actor/comunidad. Para hacerlo, agregamos un parámetro de penalización de "expansión voraz", v. Este v penaliza el tamaño del actor de una manera no lineal, de modo que cuanto más grande se vuelve, más difícil tiene moverse. La penalización se calcula como $\frac{1}{m_x^v}$ para cada actor x. Esta idea que se usa aquí para afrontar el tamaño es lo que hemos llamado la "política romana inicial", ya que identificamos este fenómeno como la dificultad de gestionar comunidades grandes, y que necesitan ser gobernadas de formas distintas cuando alcanzan cierta masa crítica, como observó César Augusto al negarse a añadir terrenos nuevos a su imperio tras vencer a los Partos [21].

El producto de las masas y A_C se multiplican. Esto es importante para la velocidad de cálculo porque el producto en el retículo [0,1] es siempre menor o igual que el mínimo. Si $A_C(x, y) = 0$, no necesitamos calcular la fuerza de atracción entre x y y porque sabemos que debe ser 0. Como la densidad de las matrices de afinidad suele ser muy baja, esto supone un gran ahorro computacional.

Como resultado, la fórmula final para la fuerza de atracción es la siguiente:

$$\mathbf{F}_{xy} = \frac{(m_x m_y)^c, A_C(x, y)}{m_x^v}, \frac{\mathbf{s}_x - \mathbf{s}_y}{|\mathbf{s}_x - \mathbf{s}_y|^3} dt \qquad (2)$$

donde A_C representa la función de afinidad elegida y $\mathbf{s}_x$ el vector de influencia del actor x.

IV-D. Escogiendo una configuración como resultado final

En el algoritmo gravitacional original, la configuración con la vida más larga en el tiempo simulado se elige como resultado. Este criterio es extremadamente rápido de calcular, pero tiende a devolver un número muy pequeño de comunidades. Esto sucede porque en los últimos pasos del algoritmo solo tenemos un número reducido de partículas que se mueven muy lentamente porque son muy pesadas.

Eso no supone un problema si el número deseado de comunidades es bajo (< 5) pero en caso de querer más, es necesario utilizar otro término para medir la calidad, R, de cada configuración, Z:

$$R(Z) = TiempoSim(Z) log(NumCom(Z)) \qquad (3)$$

Al usar esta fórmula, premiamos tanto la estabilidad en el tiempo como un mayor número de comunidades diferentes. También es posible especificar el número exacto de comunidades deseadas.

V. Comparación con otros algortimos de detección de comunidades

Por último, para comparar nuestro algoritmo de detección de la comunidad, hemos escogido datos de redes con etiquetas reales sobre las comunidades formadas: la famosa red social del club de karate de Zachary [22], Poolboks [23], que contiene la cantidad de compras conjuntas de diferentes libros sobre política estadounidense, Football [10], que representa el número de partidos entre cada par de equipos, y Dolphins [24] es una red que registra la frecuencia con la que cada par de delfines jugó juntos.

Los otros algoritmos de detección de comunidades probados son: Girvan-Newman [25], optimización de modularidad voraz de Newman [13], el algoritmo de Lovaina [14], utilizar los valores propios de la red para detectar comunidades [26] y propagación de etiquetas [27]. Hemos utilizado Normalized Mutual Information (NMI) [28] y el Adjusted Random Index (ARI) [29] para comparar los resultados.

Los resultados se muestran en la Figura 2. En la figura se muestra como el Borgia Clustering superó al resto de algoritmos en la medida conjunta de ARI+NMI, y también cada una individualmente.

VI. Conclusiones y líneas futuras

En este trabajo hemos propuesto un nuevo conjunto de funciones basadas en la naturaleza humana para representar a los actores de una red social y hemos mostrado los efectos que diferentes afinidades tienen en *República* de Platón. También hemos discutido la importancia de la escala en las redes y hemos aplicado nuestras ideas para desarrollar un nuevo algoritmo de detección de comunidades basado en un proceso histórico real, el Borgia Clustering, tomando como base las guerras centro-italianas del siglo XV. Utilizando este algoritmo, hemos obtenido buenos resultados en comparación con los algoritmos de detección de comunidades más utilizados hasta la fecha.

Creemos que este algoritmo puede ofrecer mejoras significativas en redes muy densas, cuando la modularidad no es una medida confiable para optimizar. En esos casos la simulación multiagente escala bien al nivel de complejidad necesario para modelar las interacciones de la red. Además, este algoritmo siempre converge al mismo valor, por lo que no es necesario ejecutarlo varias veces para obtener un resultado válido. Además, debido a los principios intuitivos sobre los que se basa el algoritmo, creemos que los resultados obtenidos con él pueden ser fácilmente interpretables.

La investigación futura tendrá como objetivo aplicar Borgia Clustering en redes a gran escala, estudiando las limitaciones y requisitos para realizar el proceso de detección de la comunidad de manera eficiente, en términos de memoria y tiempo de ejecución.

VII. Agradecimientos

El trabajo de Javier Fumanal Idocin y Humberto Bustince ha sido financiado por el proyecto PID2019-108392GB-I00 (AEI/10.13039/ 501100011033).

El trabajo de Oscar Cordón ha sido financiado por el gobierno de España, EXASOCO (PGC2018-101216-B-I00), incluyendo fondos de desarrollo regional europeo (ERDF).

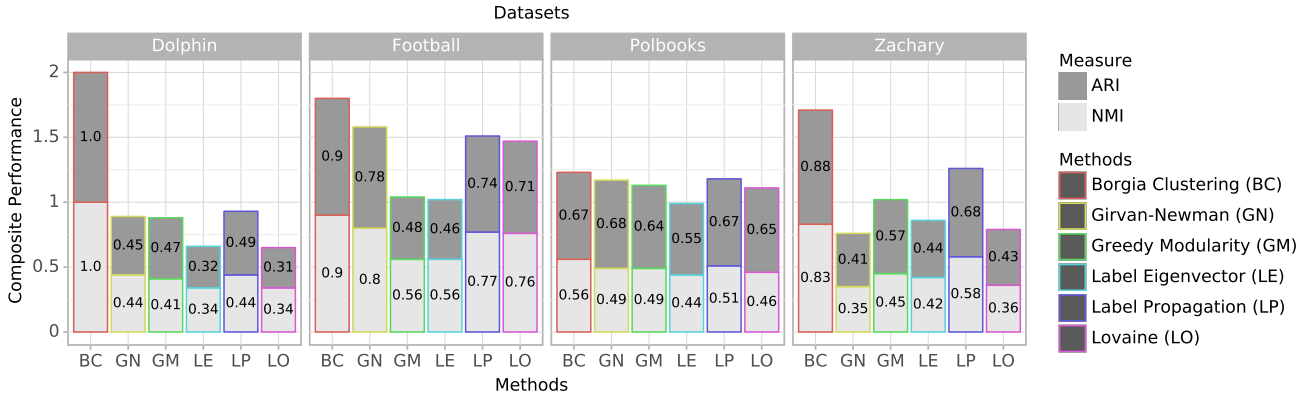


Figura 2: Comparación de diferentes algoritmos de detección de comunidades. Hemos comparado el algoritmo Borgia Clustering con tres métodos de optimización de modularidad: Girvan-Newman, optimización de modularidad voraz de Newman y el algoritmo Lovaina. También lo hemos comparado con el uso de valores propios de matrices para detectar comunidades y propagación de etiquetas.

La investigación de Amparo Alonso Betanzos ha sido parcialmente financiado por el Ministerio de Economía y Competitividad de España (TIN2015-65069-C2-1-R), por los fondos europeos FEDER y por la Consellería de Industria de la Xunta de Galicia (GRC2014 /035).

El trabajo de María Minárová ha sido financiado por los proyectos APVV-17-0066, and APVV-18-0052.

Referencias

[1] John Scott. Social network analysis. *Sociology*, 22(1):109–127, 1988.

[2] Stanley Wasserman, Katherine Faust, et al. *Social network analysis: Methods and applications*, volume 8. Cambridge university press, 1994.

[3] Stephen P Borgatti, Ajay Mehra, Daniel J Brass, and Giuseppe Labianca. Network analysis in the social sciences. *Science*, 323(5916):892–895, 2009.

[4] Steve Horvath. *Weighted network analysis: applications in genomics and systems biology*. Springer Science & Business Media, 2011.

[5] Ilan Lobel and Evan Sadler. Information diffusion in networks through social learning. *Theoretical Economics*, 10(3):807–851, 2015.

[6] Julian McAuley and Jure Leskovec. Discovering social circles in ego networks. *ACM Transactions on Knowledge Discovery from Data*, 8(1):4, 2014.

[7] Liat Sless, Noam Hazon, Sarit Kraus, and Michael Wooldridge. Forming k coalitions and facilitating relationships in social networks. *Artificial Intelligence*, 259:217 – 245, 2018.

[8] Wei Yu and Shijun Li. Recommender systems based on multiple social networks correlation. *Future Generation Computer Systems*, 87:312 – 327, 2018.

[9] Jordi Delgado. Emergence of social conventions in complex networks. *Artificial Intelligence*, 141(1):171 – 185, 2002.

[10] M. Girvan and M. E. J. Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826, Jun 2002.

[11] Haijun Zhou. Distance, dissimilarity index, and network community structure. *Physical review E*, 67(6):061901, 2003.

[12] Gergely Palla, Imre Derényi, Illés Farkas, and Tamás Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435(7043):814, 2005.

[13] Mark EJ Newman. Fast algorithm for detecting community structure in networks. *Physical review E*, 69(6):066133, 2004.

[14] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10), oct 2008.

[15] Leon Danon, Albert Diaz-Guilera, Jordi Duch, and Alex Arenas. Comparing community structure identification. *Journal of Statistical Mechanics: Theory and Experiment*, 2005(09):P09008, 2005.

[16] Mark EJ Newman and Michelle Girvan. Finding and evaluating community structure in networks. *Physical review E*, 69(2):026113, 2004.

[17] W.E. Wright. Gravitational clustering. *Pattern Recognition*, 9(3):151 – 166, 1977.

[18] Mathieu Jacomy, Tommaso Venturini, Sebastien Heymann, and Mathieu Bastian. Forceatlas2, a continuous graph layout algorithm for handy network visualization designed for the gephi software. *PLOS ONE*, 9(6):1–12, 06 2014.

[19] José Catalàn Deus. El príncipe del renacimiento. *Debate*, 26:90, 2008.

[20] Robert K Merton. The Matthew effect in science: The reward and communication systems of science are considered. *Science*, 159(3810):56–63, 1968.

[21] Alejandro Bancalari. *Orbe romano e imperio global: la romanización desde Augusto a Caracalla*. Editorial Universitaria de Chile, 2007.

[22] Wayne W Zachary. An information flow model for conflict and fission in small groups. *Journal of Anthropological Research*, 33(4):452–473, 1977.

[23] Lada A Adamic and Natalie Glance. The political blogosphere and the 2004 us election: divided they blog. In *Proceedings of the 3rd international workshop on Link discovery*, pages 36–43. ACM, 2005.

[24] David Lusseau, Karsten Schneider, Oliver J Boisseau, Patti Haase, Elisabeth Slooten, and Steve M Dawson. The bottlenose dolphin community of doubtful sound features a large proportion of long-lasting associations. *Behavioral Ecology and Sociobiology*, 54(4):396–405, 2003.

[25] Michelle Girvan and Mark EJ Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826, 2002.

[26] M. E. J. Newman. Finding community structure in networks using the eigenvectors of matrices. *Physical Review E*, 74(3):036104, Sep 2006.

[27] Usha Nandini Raghavan, Réka Albert, and Soundar Kumara. Near linear time algorithm to detect community structures in large-scale networks. *Physical review E*, 76(3):036106, 2007.

[28] Alexander Strehl and Joydeep Ghosh. Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *Journal of Machine Learning Research*, 3(Dec):583–617, 2002.

[29] William M Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association*, 66(336):846–850, 1971.

Evaluación de modelos multilingües pre-entrenados en similitud semántica para la lucha contra la desinformación

Álvaro Huertas-García*†, Alejandro Martín*, Javier Huertas-Tato*, David Camacho*
*Departamento de Sistemas Informáticos, Universidad Politécnica de Madrid, Madrid, Spain
Email: alvaro.huertas.garcia@alumnos.upm.es, {alejandro.martin,javier.huertas.tato,david.camacho}@upm.es
†Departamento de Informática, Universidad Rey Juan Carlos, Móstoles, Spain

***Resumen*—La presencia de desinformación y contenidos nocivos en las redes sociales es un problema emergente que pone en peligro la salud pública. Una de las estrategias para detectar, evaluar y dar una respuesta rápida a este problema de desinformación, son las técnicas de Procesamiento del Lenguaje Natural (PLN) basadas en similitud semántica. A la hora de combatir la desinformación, el idioma es una de las mayores barreras a las que hay que hacer frente, lo que denota la necesidad de desarrollar herramientas multilingües para responder de forma eficaz a este problema. En este trabajo, evaluamos modelos multilingües pre-entrenados basados en Transformadores entrenados mediante una arquitectura de red siamesa. Estos modelos se aplican a tareas de similitud textual semántica (STS) a partir de la ampliación del STS Benchmark (STSb) a 15 idiomas. Nuestros resultados desglosan el rendimiento de los modelos multilingües pre-entrenados en función del idioma, y la utilidad de la extensión del STS Benchmark a un nivel multilingüe. Estos resultados constituyen un primer paso hacia la extensión de métodos propuestos en la literatura que emplean la similitud semántica para combatir la desinformación a un nivel multilingüe.**

***Palabras clave*—Desinformación, Procesado del Lenguage Natural, Transformer, Similitud Semántica, Multilingualidad**

I. Introducción

La tecnología y las redes sociales nos mantienen informados, productivos y conectados. Sin embargo, estas plataformas también permiten y amplifican la rápida difusión de información manipuladora y falsa. En la literatura el término *fake news* se ha utilizado tradicionalmente para referirse a esta situación [5], [10]. Sin embargo, no existe una definición clara y aceptada. Un grupo creciente de expertos en medios de comunicación e instituciones, como el Instituto Poynter[1] y el Consejo Europeo [10], proponen dejar de utilizar el término *fake news* y reemplazarlo por el término *desinformación*, ya que éste describe en mejor medida la complejidad del ecosistema del desorden informativo [10]. Siguiendo estas recomendaciones, en este trabajo se empleará el término desinformación.

Cabe señalar que en el vocabulario inglés se diferencian dos términos para explicar la desinformación: *disinformation* y *misinformation*. Ambos términos se refieren al fenómeno de contrarrestar, socavar y perjudicar la calidad de la información mediante la incorporación de información engañosa. La diferencia entre estos términos reside en el propósito o motivo de la información engañosa. Mientras que el término *misinfomation* se refiere a un propósito involuntario, el término *disinformation* se emplea en el caso en el que el objetivo es engañar a la gente intencionadamente. En este trabajo nos referiremos a ambas situaciones empleando el término desinformación.

Desde que el COVID-19 (la enfermedad infecciosa causada por el virus del SARS-CoV-2) surgió en Wuhan (China) en diciembre de 2019, el público ha sido bombardeado con enormes cantidades de información, gran parte de la cual no se comprueba. La desinformación afecta a la respuesta sanitaria para la gestión de enfermedades, lo que ha llevado a la Organización Mundial de la Salud (OMS) a acuñar esta situación con el término *infodemia* [6], [22].

Las redes sociales han sido identificadas como las mejores fuentes para controlar la desinformación y combatir los rumores, el estigma y las teorías conspirativas entre la gente [6]. La gran cantidad de datos presente en estas plataformas dificulta la detección y respuesta a esta infodemia. Es por eso que la automatización de estos procesos es necesaria y puede llevarse a cabo mediante herramientas del Procesamiento del Lenguaje Natural (PLN). El PLN es un campo interdisciplinario de la Inteligencia Artificial que utiliza técnicas computacionales para el análisis y la representación automática del lenguaje humano [31]. Uno de los niveles del lenguaje de los que se ocupan los sistemas de PNL es la semántica. La semántica determina los posibles significados de una frase centrándose en las interacciones a nivel de palabra. Por lo tanto, los modelos de PNL son capaces de codificar el significado de las frases en vectores dentro de un espacio de alta dimensión [24]. Estos vectores se denominan *embeddings*.

Como se describe en [10], uno de los requisitos para identificar información fiable y útil es la presencia de un código común entre el emisor y el receptor (i.e. un idioma común). Al igual que en las lenguas habladas, no hay una distribución uniforme de los idiomas en Internet. El inglés es el idioma más utilizado, ya que el 56,8 % de los sitios web lo utilizan, mientras que el segundo y tercer idioma son el ruso y el español, con un 7,6 % y un 4,6 %, respectivamente [20]. Por consiguiente, es un hecho que el idioma determina el acceso a la información en línea y es una de las limitaciones en la

[1]https://www.poynter.org/

lucha contra la desinformación.

En este trabajo, proponemos la extensión multilingüe del reconocido Benchmark de Similitud Textual Semántica en inglés (STSb) [3]. Además, utilizamos esta versión ampliada para desglosar el rendimiento de modelos pre-entrenados multilingües basados en Transformadores en la tarea de similitud semántica en diferentes idiomas.

II. Trabajos previos y Estado del Arte

En esta sección abordamos los modelos basados en Transformadores, la importancia de los modelos multilingües y trabajos relacionados que utilizan la similitud semántica para combatir la desinformación en la web.

La aparición de modelos basados en Transformadores [28], como BERT [9] o RoBERTa [21], han establecido un nuevo estado del arte en varias tareas de PNL como la similitud semántica [24]. Tradicionalmente, estos modelos utilizan arquitecuras denominadas *cross-encoders*. En este tipo de arquitecturas se introducen dos frases en el transformador y se predice un valor objetivo [17], [24]. Esto requiere el cálculo de la interacción entre todos los términos de ambas frases, lo que provoca una enorme sobrecarga computacional al tener un crecimiento cuadrático [9]. Para superar estas dificultades, se puede condensar la semántica de una frase en embeddings mediante arquitecturas entrenadas de forma siamesa [24]. Como muestra la Figura 1, las arquitecturas siamesas consisten en dos modelos basados en Transformadores ligados que pueden ser ajustados en una tarea específica como el cálculo de las puntuaciones de similitud. Este enfoque también se denomina *dual encoders* o *bi-encoders* [18].

A pesar de que los modelos basados en Transformadores hayan marcado un antes y un después en el estado del arte del campo del PLN, siguen sin solventar el cuello de botella que supone el idioma. La mayoría de los modelos son monolingües, normalmente siendo el inglés la lengua para la que han sido entrenados dado que es un idioma con alto nivel de recursos [25]. Ampliar los modelos existentes a nuevas lenguas con un nivel de recursos inferior es un nuevo reto en PNL. Los modelos multilingües aprenden a mapear el texto de varios idiomas a un espacio de vectores compartido por los diferentes idiomas (o espacio multilingüe) utilizando datos paralelos [27]. Como resultado, en este espacio vectorial, las palabras relacionadas o similares estarán más cerca unas de otras, y las no relacionadas estarán distantes, independientemente del idioma. XML-R [7], LaBSE [11] y mUSE [31] son algunos ejemplos de modelos obtenidos siguiendo este procedimiento.

En lo que se refiere a la transferencia de conocimiento entre sistemas, otra estrategia para desarrollar modelos multiligües consiste en transferir el conocimiento de modelos monolingües entrenados en lenguas de altos recursos y ajustados a una tarea concreta a lenguas de recursos medios o bajos [25], [26]. Reimers y Gurevych [25], utilizaron una técnica de destilación de conocimiento denominada estrategia profesor-estudiante [2], [16] para crear modelos multilingües a partir de modelos monolingües. Esta estrategia emplea un modelo monolingüe (i.e. profesor) para generar embeddings a partir de datos en el idioma de origen. A continuación, estos embeddings se emplean para entrenar un modelo multilingüe (i.e. estudiante) que ha de generar embeddings similares con datos procedente de los idiomas a los que se quiere extender el conocimiento.

Actualmente, en el aprendizaje automático, la cantidad de datos de entrenamiento disponibles, junto a la complejidad del modelo, es uno de los principales factores que influyen en el rendimiento de un algoritmo [1]. Asimismo, la escasez de datos multilingües dificulta el desarrollo de modelos multilingües entrenados para tareas específicas. En la literatura encontramos autores que previamente han intentado ampliar dataset monolingües a otras lenguas. Un ejemplo ilustrativo es el intento realizado por Ham et al. [15] para traducir el STS Benchmark (STSb) inglés, entre otros conjuntos de datos, al coreano. Asimismo, Reimers y Gurevych [25] utilizaron Google Translator para ampliar la tarea de 2017 del STSb a los idiomas alemán, francés, italiano y holandés. Hasta donde sabemos, nadie ha extendido completamente el STSb, un conjunto de datos en inglés utilizados en las tareas STS entre 2012 y 2017, a un escenario multilingüe.

En relación a investigaciones previas sobre la lucha contra la desinformación, Jwa et al. [19] emplearon por primera vez un modelo basado en BERT para la detección de desinformación, superando los resultados de técnicas anteriores. Igualmente, en la literatura se han propuesto estrategias basadas en la similitud semántica para la detección de desinformación. Vijjali et al. [29], proponen el uso de un proceso automatizado en dos pasos. El primer paso consiste en recuperar las afirmaciones más relevantes de una base de datos de información contrastada sobre la COVID-19 utilizando la similitud semántica y, en segundo lugar, evaluar la concordancia entre la afirmación contrastada seleccionada y los hechos a analizar. Una estrategia similar se describe en [12], donde los autores presentan un enfoque para analizar automáticamente la credibilidad de la información de los mensajes propagados a través de WhatsApp, comparando estos mensajes con artículos de noticias utilizando la similitud semántica. Guo et al. [14] introdujeron el conjunto de datos CORD19STS, que incluye 13.710 pares de frases anotadas recopiladas del conjunto de datos CORD-19[2] para lograr una tarea STS específica del dominio COVID-19. Otro uso de la similitud semántica es BERTopic [13], una técnica de clasificación no supervisada de temas que utiliza la distancia coseno entre los embeddings de las frases para agruparlas y extraer los temas y las palabras más representativas por clúster. De este modo se obtiene información sobre cómo está estructurado un conjunto de datos de documentos. Estas aplicaciones se basan en la medición de la similitud textual semántica (STS), lo que convierte a la STS en una tarea crucial para combatir la desinformación.

III. Modelos multilingües pre-entrenados

Los modelos bi-encoders multilingües pre-entrenados evaluados en la versión multilingüe ampliada de STS Benchmark

[2]https://www.kaggle.com/allen-institute-for-ai/CORD-19-research-challenge

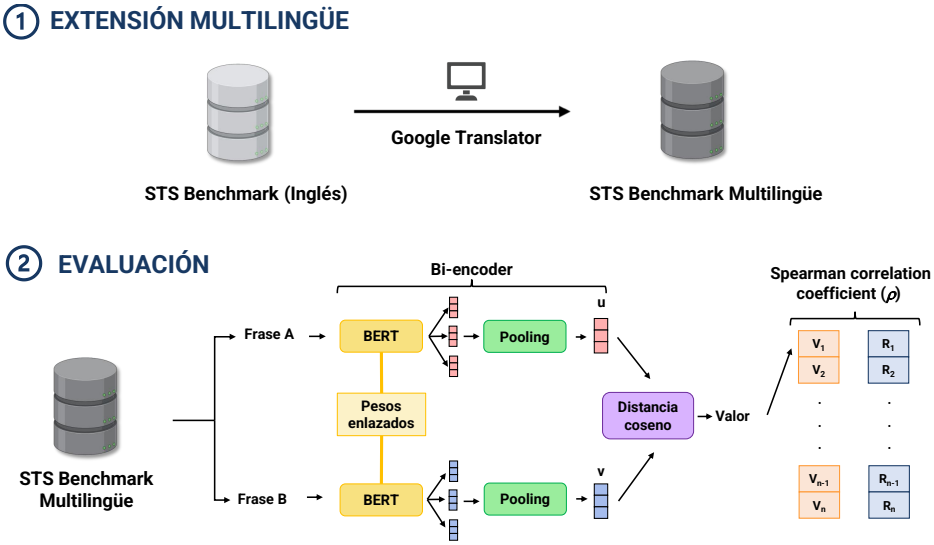


Figura 1: Metodología aplicada para ampliar el dataset STS Benchmark a 15 idiomas y la representación del pre-entrenamiento en forma de red siamesa de los modelos de multilingües en la tarea de similitud textual semántica. Las frases de una pareja de frases etiquetadas con el grado de similitud entran por separado en el modelo. Tras obtener los embeddings de las frases, la puntuación de similitud semántica se calcula con la distancia coseno. Por último, el rendimiento se calcula mediante el coeficiente de correlación de Spearman (ρ) entre las puntuaciones predichas y los valores reales.

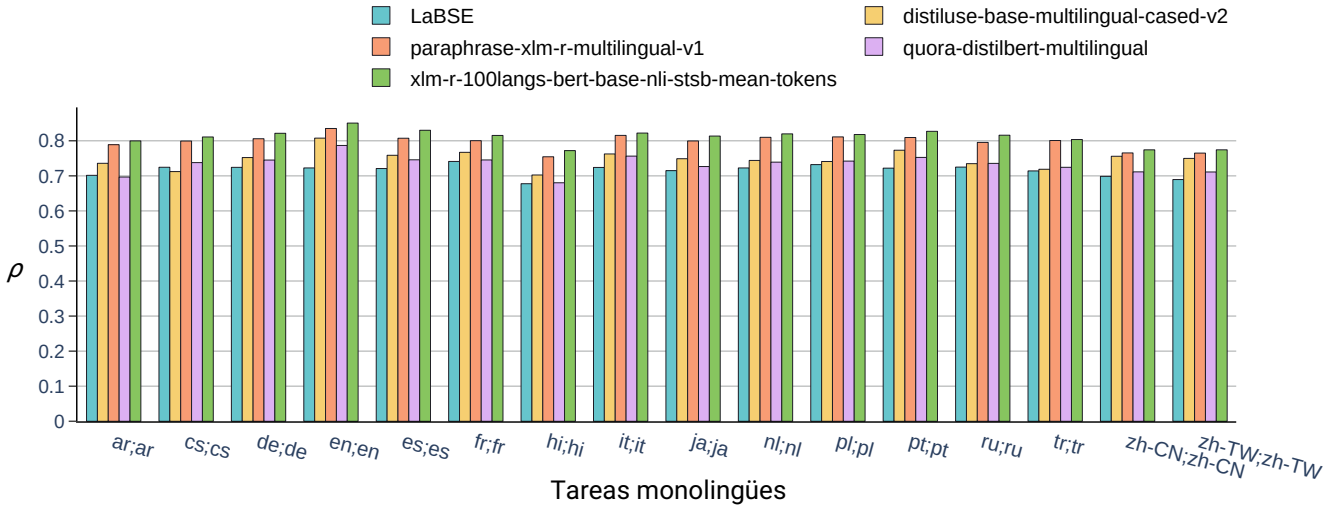
fueron extraídos de SentenceTransformers [24], [25]:

- **Modelo 1 - *distiluse-base-multilingual-cased-v2***: Versión destilada del modelo universal de oraciones multilingüe (mUSE) [4], [31] ampliada para soportar 50+ idiomas. mUSE es un transformador bi-encoder con entrenamiento multitarea en tareas de inferencia del lenguaje natural (NLI) y de respuesta a preguntas. Para alinear los espacios vectoriales multilingües, mUSE utiliza una tarea de clasificación de traducciones. Dado un par de traducciones (s_i, t_i) y varias traducciones alternativas incorrectas, el modelo tiene que identificar la traducción correcta.
- **Modelo 2 - *LaBSE***: Modelo *Language-agnostic BERT Sentence Embedding* [11] entrenado de forma similar a mUSE con una arquitectura siamesa basada en BERT y 6.000 millones de parejas de frases en 109 idiomas.
- **Modelo 3 - *paraphrase-xlm-r-multilingual-v1***: Versión destilada multilingüe mediante la técnica profesor-estudiante de RoBERTa [21] entrenada en datos de paráfrasis a gran escala. El modelo profesor monolingüe es el modelo RoBERTa entrenado en datos de paráfrasis a gran escala y XLM-R [7] el modelo estudiante multilingüe. XML-R está basado en el modelo RoBERTa que dispone de un vocabulario de mayor tamaño y entrenado en un amplio conjunto de datos multilingües.
- **Modelo 4 - *xlm-r-100langs-bert-base-nli-stsb-mean-tokens***: Versión destilada multilingüe de BERT [9] entrenada en NLI [30] y STSb [3] usando XLM-R como modelo de estudiante.
- **Modelo 5 - *quora-distilbert-multilingual***: Versión destilada usando el modelo monolingüe BERT y la versión multilingüe BERT, como profesor y estudiante, respectivamente. El modelo BERT monolingüal se encuentra entrenado en NLI y STSb, y la detección de preguntas duplicadas de Quora.

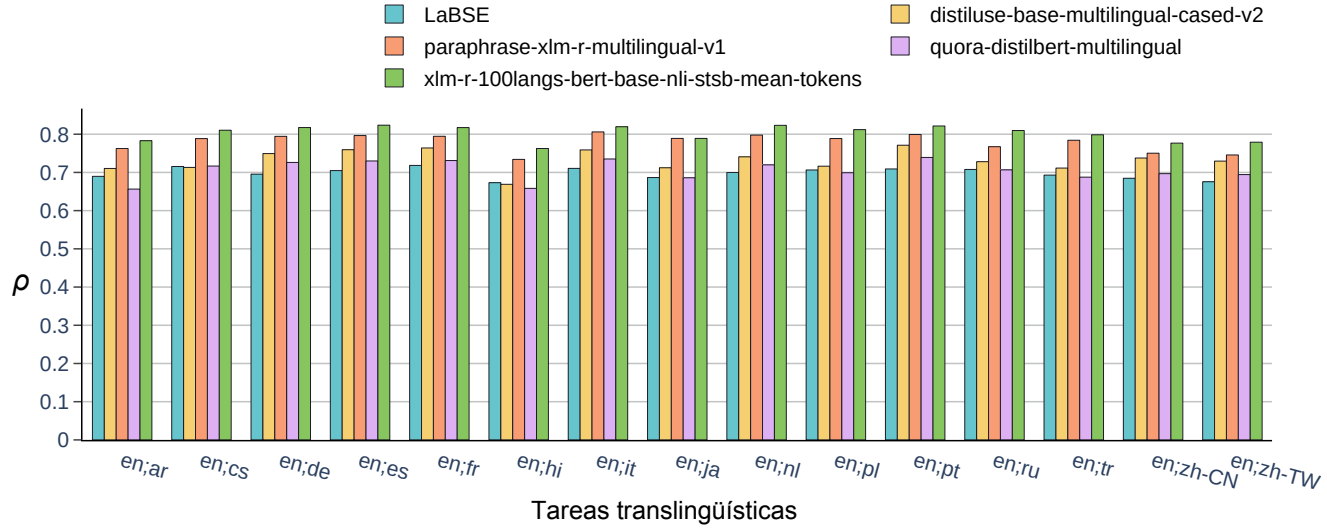
IV. Extensión multilingüe del STS Benchmark

El STS Benchmark original [3] consta de particiones de entrenamiento, desarrollo y test de 5749, 1500 y 1379 pares de frases, respectivamente, etiquetadas con una puntuación de similitud entre 0 y 5, de menos a más similar. Con el fin de ampliar el STSb original en inglés a un conjunto de datos multilingües, traducimos las particiones del STS Benchmark del inglés a 15 idiomas[3]. Como se puede ver en la Figura 1, el paquete python del Traductor de Google[4] se utiliza para este propósito. El uso del Traductor de Google se justifica por investigaciones presentes en la literatura donde también emplean esta herramienta [15], [25]. Cabe señalar que el Traductor de Google distingue dos variantes del chino: simplificado y con términos del chino continental (*zh-CN*), y tradicional y con términos del taiwanés (*zh-TW*). Se descartaron los pares de frases traducidas con un valor de confianza inferior a 0,7. Como resultado, el neerlandés es el idioma con menos pares de frases en los conjuntos de desarrollo (1483 pares de frases) y de test (1358 pares de frases). El STS Benchmark multilingüe está disponible públicamente en GitHub[5].

[3] ar, cs, de, en, es, fr, hi, it, ja, nl, pl, pt, ru, tr, zh-CN, zh-TW
[4] https://pypi.org/project/google-trans-new/
[5] https://github.com/Huertas97/Multilingual-STSB



(a)



(b)

Figura 2: Coeficiente de correlación de Spearman ρ en las tareas monolingües (a) y translingüísticas (b) procedentes de la partición de test de la versión multilingüe del STSb entre entre los valores reales y la similitud calculada por los modelos multilingües.

V. Evaluación - Similitud Semántica Multilingüe

Para medir la similitud semántica en STSb, se computan los embeddings de cada par de frases y se mide la similitud semántica mediante la métrica de similitud del coseno. Seguidamente, para evaluar el rendimiento en la tarea de similitud semántica, se calcula el coeficiente de correlación de Spearman entre las puntuaciones obtenidas y las puntuaciones reales (véase la Figura 1). Como se menciona en [8], el coeficiente de correlación de rangos de Spearman no es paramétrico, ya que sustituye las observaciones por su rango y calcula la correlación. Por lo tanto, se selecciona como métrica de evaluación dado que es insensible a los valores atípicos, las relaciones no lineales o los datos que no siguen una distribución normal [23].

Cabe señalar que, además de la versión traducida automáticamente de STS Benchmark dividida en 15 idiomas, como se ha descrito anteriormente, los idiomas también se han combinado en tareas monolingües y multilingües, lo que da un total de 31 tareas. Las tareas monolingües tienen ambas frases de la misma fuente lingüística (por ejemplo, Ar-Ar, Es-Es), mientras que las tareas translingüísticas tienen dos frases, cada una en un idioma diferente siendo una de ellas la inglesa (por ejemplo, En-Ar, En-Es). La Figura 2 representa el desglose del rendimiento del STSb multilingüe según las tareas monolingües y translingüísticas. A partir de estos resultados,

Cuadro I: Coeficiente de correlación de Spearman ($\rho \times 100$) entre las etiquetas reales para la partición test del STSb ampliado y la similitud calculada por modelos bi-encoders multilingües.

Model	EN-EN	ES-ES	EN-ES	Avg. Monolingüe	Avg. Translingüistica	Avg.
Modelo 1	78.74	75.86	75.94	74.88	73.25	74.10
Modelo 2	83.50	80.72	79.65	79.84	78.08	79.01
Modelo 3	85.04	82.99	82.36	81.14	80.37	80.77
Modelo 4	72.25	72.08	70.48	71.62	69.83	70.78
Modelo 5	78.66	74.57	73.00	73.44	70.64	72.12

Cuadro II: Coeficiente de correlación de Spearman ($\rho \times 100$) entre las etiquetas reales para la partición test del STSb ampliado y la similitud calculada por modelos bi-encoders y cross-encoders monolingües .

	Model	EN-EN
Bi-encoders Monolingües	roberta-large-nli-stsb-mean-tokens	86.39
	roberta-base-nli-stsb-mean-tokens	85.44
	bert-large-nli-stsb-mean-tokens	85.27
	bert-base-nli-stsb-mean-tokens	85.05
Cross-encoders Monolingües	ce-roberta-large-stsb	91.47
	ce-roberta-base-stsb	90.17

podemos ver cómo el Modelo 4 (*xlm-r-100langs-bert-base-nli-stsb-mean-tokens*) seguido del Modelo 3 (*paraphrase-xlm-r-multilingual-v1*) tienen los mejores resultados. Como era de esperar, el mejor resultado se obtiene en la tarea monolingüe original en inglés. Resulta interesante comprobar la capacidad multilingüe de los modelos, ya que las lenguas cercanas al inglés (por ejemplo, el holandés, el alemán y el francés) y algunas más distantes (por ejemplo, el japonés, el ruso y el polaco) obtienen una puntuación superior a $0,8$. Aunque el rendimiento no es ideal en los idiomas chino e hindi, creemos que nuestros resultados demuestran que el multilingüismo puede utilizarse para la similitud semántica y ampliar a nuevos idiomas las técnicas de lucha contra la desinformación basadas en ella.

El rendimiento de las tareas translingüísticas se muestra en la Figura 2b. En comparación con las tareas monolingües, el rendimiento disminuye. Este caso se ilustra en la Tabla I. Por razones de espacio, sólo mostramos los resultados de inglés, español y la media de los resultados en las tareas monolingües y translingüísticas, y en todos los idiomas. En esta tabla se observa como todos los modelos disminuyen la media del coeficiente de correlación de Spearman en las tareas translingüísticas. Cabe destacar que la media de los coeficientes de correlación se calcula transformando cada coeficiente de correlación en un valor z de Fisher, promediándolos y volviéndolos a transformar en un coeficiente de correlación. Además de los modelos bi-encoders multilingües, se evalúan modelos bi-encoders y cross-encoders monolingües en la STSb original en inglés (véase la tabla II). Como se preveía, los cross-encoders obtienen grandes resultados con un mayor coste computacional. No obstante, los dos mejores modelos multilingües, Modelo 4 (*xlm-r-100langs-bert-base-nli-stsb-mean-tokens*) y Modelo 3 (*paraphrase-xlm-r-multilingual-v1*), demuestran que pueden competir con los bi-encoders monolingües.

VI. Conclusiones y Trabajo Futuro

En este trabajo hemos evaluado y desglosado el rendimiento de los modelos multilingües en los diferentes idiomas empleando una versión extendida del reconocido Benchmark de Similitud Textual Semántica en inglés (STSb).

De la investigación realizada se puede concluir que dos de los cinco modelos multilingües evaluados, *xlm-r-100langs-bert-base-nli-stsb-mean-tokens* y *paraphrase-xlm-r-multilingual-v1*, pueden alcanzar un rendimiento similar al de los bi-encoders monolingües en las tareas de Similitud Semántica Textual (STS). Estos resultados indican que aún hay margen de mejora para los modelos multiligües, pudiendo ser mejorados mediante el entrenamiento con datos multilingües específicos de la tarea de similitud semántica. Igualmente, se ha demostrado que los cross-encoders obtienen mejores resultados que los bi-encoders a costa del coste computacional.

Los modelos multilingües tienen la ventaja de tener un impacto más significativo en la lucha contra la desinformación al introducir el multilingüismo y superar el cuello de botella lingüístico. Según los resultados de la evaluación, los cinco modelos multilingües utilizados en la investigación pueden emplearse en tareas de similitud textual semántica monolingües y translingüísticas, con un mejor rendimiento en tareas monolingües y en idiomas cercanos al inglés como el español, el holandés o el alemán. No obstante, estos resultados también muestran que los modelos multilingües pueden tratar con lenguas no cercanas al inglés, como el ruso o el japonés.

La versión ampliada multilingüe propuesta en este trabajo para las particiones de entrenamiento, desarrollo y test del Benchmark de Similitud Semántica Textual puede utilizarse en la práctica para extender a nuevos idiomas herramientas monolingües que luchen contra la desinformación, como las descritas en la Sección II. Los resultados sugieren que este conjunto de datos ampliado también podría ser útil para entrenar modelos con datos multilingües específicos para la tarea de similitud semántica. Será necesario seguir investigando para validar la utilidad de los datos de entrenamiento multilingües presentados en este artículo. Este trabajo abre las puertas a futuras investigaciones en la extensión de datos monolingües a otros idiomas, siendo el conjunto de datos CORD19STS un buen ejemplo.

Agradecimientos

Esta investigación está financiada por el proyecto CIVIC: Caracterización inteligente de la veracidad de la información relacionada con COVID-19, concedido por la FUNDACIÓN BBVA SUBVENCIONES PARA EQUIPOS DE INVESTIGACIÓN CIENTÍFICA SARS-CoV-2 y COVID-19.

Esta investigación ha sido apoyada por el Ministerio de Ciencia y Educación bajo el proyecto TIN2017-85727-C4-3-P (DeepBio), por el Proyecto PACMEL de la BDSI CHIST-ERA 2017 (PCI2019-103623), por la Comunidad de Madrid

bajo la beca S2018/ TCS-4566 (CYNAMON), S2017/BMD-3688, y por la Comisión Europea, proyecto 2020-EU-IA-0252 IBERIFIER - Iberian Digital Media Research and Fact-Checking Hub. Acción parcialmente financiada por la Comunidad de Madrid en el marco del Convenio Plurianual con la Universidad Politécnica de Madrid en la línea de actuación Programa de Excelencia para el Profesorado Universitario.

Referencias

[1] Bishop, C.M.: Pattern recognition and machine learning. Information science and statistics, Springer, New York (2006)
[2] Buciluundefined, C., Caruana, R., Niculescu-Mizil, A.: Model compression. p. 535–541. KDD '06, Association for Computing Machinery, New York, NY, USA (2006). https://doi.org/10.1145/1150402.1150464, https://doi.org/10.1145/1150402.1150464
[3] Cer, D., Diab, M., Agirre, E., Lopez-Gazpio, I., Specia, L.: SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In: Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017). pp. 1–14. Association for Computational Linguistics, Vancouver, Canada (Aug 2017). https://doi.org/10.18653/v1/S17-2001
[4] Chidambaram, M., Yang, Y., Cer, D., Yuan, S., Sung, Y., Strope, B., Kurzweil, R.: Learning cross-lingual sentence representations via a multi-task dual-encoder model. In: Proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP-2019). pp. 250–259. Association for Computational Linguistics, Florence, Italy (2019). https://doi.org/10.18653/v1/W19-4330
[5] Chowdhury, G.G.: Natural language processing. Annual Review of Information Science and Technology **37**(1), 51–89 (2005). https://doi.org/10.1002/aris.1440370103
[6] Cinelli, M., Quattrociocchi, W., Galeazzi, A., Valensise, C.M., Brugnoli, E., Schmidt, A.L., Zola, P., Zollo, F., Scala, A.: The COVID-19 social media infodemic. Scientific Reports **10**(1), 16598 (2020). https://doi.org/10.1038/s41598-020-73510-5
[7] Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., Grave, E., Ott, M., Zettlemoyer, L., Stoyanov, V.: Unsupervised cross-lingual representation learning at scale (2020)
[8] Dalgaard, P.: Introductory statistics with r. Statistics and Computing, Springer New York, New York, NY (2008). https://doi.org/10.1007/978-0-387-79054-1
[9] Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding (2019)
[10] Estrada-Cuzcano, A., Alfaro-Mendives, K., Saavedra-Vásquez, V.: Disinformation y misinformation, posverdad y fake news: precisiones conceptuales, diferencias, similitudes y yuxtaposiciones. Información, cultura y sociedad (42), 93–106 (2020). https://doi.org/10.34096/ics.i42.7427
[11] Feng, F., Yang, Y., Cer, D., Arivazhagan, N., Wang, W.: Language-agnostic bert sentence embedding (2020)
[12] Gaglani, J., Gandhi, Y., Gogate, S., Halbe, A.: Unsupervised whatsapp fake news detection using semantic search. In: 2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS). pp. 285–289 (2020). https://doi.org/10.1109/ICICCS48265.2020.9120902
[13] Grootendorst, M.: Bertopic: Leveraging bert and c-tf-idf to create easily interpretable topics. (2020). https://doi.org/10.5281/zenodo.4430182
[14] Guo, X., Mirzaalian, H., Sabir, E., Jaiswal, A., Abd-Almageed, W.: Cord19sts: Covid-19 semantic textual similarity dataset (2020)
[15] Ham, J., Choe, Y.J., Park, K., Choi, I., Soh, H.: Kornli and korsts: New benchmark datasets for korean natural language understanding (2020)
[16] Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network (2015)
[17] Humeau, S., Shuster, K., Lachaux, M.A., Weston, J.: Poly-encoders: Transformer architectures and pre-training strategies for fast and accurate multi-sentence scoring (2020)
[18] Humeau, S., Shuster, K., Lachaux, M.A., Weston, J.: Poly-encoders: Transformer architectures and pre-training strategies for fast and accurate multi-sentence scoring (2020)
[19] Jwa, H., Oh, D., Park, K., Kang, J.M., Lim, H.: exbake: Automatic fake news detection model based on bidirectional encoder representations from transformers (bert). Applied Sciences **9**(19) (2019). https://doi.org/10.3390/app9194062
[20] Kemp, S.: Digital 2020: october global statshot (2020), https://datareportal.com/reports/digital-2020-october-global-statshot
[21] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: Roberta: A robustly optimized bert pretraining approach (2019)
[22] Naeem, S.B., Bhatti, R.: The Covid-19 'infodemic': a new front for information professionals. Health Information & Libraries Journal **37**(3), 233–239 (2020). https://doi.org/10.1111/hir.12311
[23] Reimers, N., Beyer, P., Gurevych, I.: Task-oriented intrinsic evaluation of semantic textual similarity. In: Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers. pp. 87–96. The COLING 2016 Organizing Committee, Osaka, Japan (2016)
[24] Reimers, N., Gurevych, I.: Sentence-bert: Sentence embeddings using siamese bert-networks (2019)
[25] Reimers, N., Gurevych, I.: Making monolingual sentence embeddings multilingual using knowledge distillation (2020)
[26] Sun, H., Wang, R., Chen, K., Utiyama, M., Sumita, E., Zhao, T.: Knowledge distillation for multilingual unsupervised neural machine translation (2020)
[27] Søgaard, A., Vulić, I., Ruder, S., Faruqui, M.: Cross-lingual word embeddings. Synthesis Lectures on Human Language Technologies **12**(2), 1–132 (2019). https://doi.org/10.2200/S00920ED2V01Y201904HLT042
[28] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need (2017)
[29] Vijjali, R., Potluri, P., Kumar, S., Teki, S.: Two stage transformer model for covid-19 fake news detection and fact checking (2020)
[30] Williams, A., Nangia, N., Bowman, S.: A broad-coverage challenge corpus for sentence understanding through inference. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers). pp. 1112–1122. Association for Computational Linguistics, New Orleans, Louisiana (Jun 2018). https://doi.org/10.18653/v1/N18-1101
[31] Yang, Y., Cer, D., Ahmad, A., Guo, M., Law, J., Constant, N., Abrego, G.H., Yuan, S., Tar, C., Sung, Y.H., Strope, B., Kurzweil, R.: Multilingual universal sentence encoder for semantic retrieval (2019)

XIV Congreso Español de Metaheurísticas, Algoritmos Evolutivos y Bioinspirados

Trabajos Destacados



Hybrid GRASP-SO-PR algorithm. The closest vs. the most balanced

1st Jesús Sánchez-Oro
Computer Science Dep.
Universidad Rey Juan Carlos
Madrid, Spain
jesus.sanchezoro@urjc.es

2nd Ana D. López-Sánchez
Economics and Quantitative Methods Dep.
Universidad Pablo de Olavide
Sevilla, Spain
adlopsan@upo.es

3rd Anna Martínez-Gavara
Statistics and Operational Research Dep.
Universitat de València
València, Spain
Ana.Martinez-Gavara@uv.es

4th Alfredo G. Hernandández-Díaz
Economics and Quantitative Methods Dep.
Universidad Pablo de Olavide
Sevilla, Spain
agarher@upo.es

5th Abraham Duarte
Computer Science Dep.
Universidad Rey Juan Carlos
Madrid, Spain
abraham.duarte@urjc.es

***Abstract*—We present a hybridization of Strategic Oscillation with Path Relinking for providing a set of high quality non-dominated solutions for the Multi-objective k-Balanced Center Location problem. The considered location problem seeks to locate k out of m facilities in order to serve n demand points, minimizing the maximum distance between any demand point and its closest facility while balancing the workload among the facilities. Computational experimentation compares the performance of our proposal, including the best method found in the state of the art as well as traditional multi-objective evolutionary algorithms.**

***Index Terms*—Discrete Optimization, k-Center Problem, k-Balanced Problem, Multi-objective Optimization, GRASP, Strategic Oscillation, Path Relinking**

I. Introduction

In this work, we would like to give community members the opportunity to learn about unfamiliar work and foster interdisciplinarity. Specifically, we will present [2], a work recently published with the following contributions: (i) to propose a novel metaheuristic method to solve the k-Balanced Center Location (k-BCL) problem, (ii) to propose a new strategy based on Path Relinking to obtain feasible solutions when two infeasible solutions are combined since the algorithm proposed is a Strategic Oscillation, and (iii) to improve the state-of-the-art algorithms for solving the k-BCL problem.

The k-BCL was first addressed in [1], considering two different functions to measure the balance: minimize the maximum number of demand points assigned to each selected facility and minimize the difference between the maximum and the minimum number of demand points assigned to each selected facility.

II. The k-Balanced Center Location Problem

The k-BCL problem can be defined as follows. Let $N = \{1, \ldots, n\}$ be a set of demand points and $M = \{1, \ldots, m\}$ be the set of candidate locations to host a facility. Let d_{ij} be the distance between the demand point $i \in N$ and the facility $j \in M$. It is assumed that all demand points are serviced/assigned to its closest facility. We define n_j as the number of demand points assigned to a facility j. The aim of the k-BCL problem is to locate a set of k out of m candidate facilities in order to minimize the maximum distance between each demand point and its closest selected facility, and balance the number of demand points assigned to each selected facility. More specifically, the former goal ensures that the distance travelled by each demand point is, at most, the obtained value, whereas the latter balances the number of demand points assigned to each selected facility, that is, to balance the workload of each selected facility.

Thus, [1] formulates the k-BCL problem as a multi-objective facility location problem where the first objective function is the one optimized in the k-Center problem, with the goal of obtaining a subset S of M with $|S| = k$ in such a way that

$$f_1(S) = \max_{i \in N} \min_{j \in S} d_{ij} \tag{1}$$

is minimized. Note that the considered problem forces that each demand point must be served by its closest facility.

In order to ensure a balance solution, [1] proposes two different functions to measure the fairness among the facilities: the first one consists of minimizing the maximum number of demand points assigned to each facility, while the second one minimizes the difference between the maximum and minimum number of demand points served by any facility. Formally:

$$f_2(S) = \max_{j \in S} n_j, \tag{2}$$

$$f_3(S) = \max_{j, j' \in S, j \neq j'} \left| n_j - n_{j'} \right|. \tag{3}$$

III. A hybrid Strategic Oscillation with Path Relinking algorithm

We propose a methodology based on Strategic Oscillation (SO), that is a metaheuristic initially designed with the purpose

of crossing back and forth between the feasible and infeasible solution spaces of an optimization problem. A feasible solution for the k-BCL problem is represented as a set S of k selected facilities to which demand points have been assigned. As previously mentioned, it is assumed that all demand points are assigned to its closest facility. Therefore, a solution is not feasible if the number of selected facilities is smaller or larger than k. To construct feasible and unfeasible solutions a GRASP method has been implemented.

GRASP is an iterative process, in which each iteration consists of two phases: a greedy randomized construction phase and a local search method. In the context of the k-BCL problem, since we are dealing with a multi-objective optimization problem with three different objective functions, we propose using each one of them separately in every construction to generate solutions focused on every objective of the problem under consideration. These greedy functions are the three considered objective functions: $f_1(S)$, $f_2(S)$ or $f_3(S)$. The local search method is based on the exchange of a selected facility $u \in S$ with a non-selected one $v \in M \setminus S$. In this phase we will have feasible solutions with k selected facilities and unfeasible solutions with $k - \beta \cdot k$ and $k + \beta \cdot k$ facilities.

Then, we hybridize the SO algorithm with a combination method based on Path Relinking (PR), in order to improve the quality of the population found. PR is based on exploring trajectories connecting pairs of solutions by generating a trajectory that link an *initiating solution* (S_i) to a *guiding solution* (S_g). In other words, PR creates a path between two solutions by iteratively including in the initiating solution attributes from the guiding one. The rationale behind this idea is that a path from the initiating solution to a guiding solution will yield new solutions that share attributes from both solutions. We propose a novel application of PR that is used as the repairing method inside the SO algorithm. In particular, PR is applied to generate a path between a pair of solutions that result from the infeasible solutions obtained in the SO. The initiating solution contains $k + \beta \cdot k$ facilities, while the guiding solution contains $k - \beta \cdot k$ facilities. Then, the path of solutions that connects them is produced by removing as many facilities in S_i to make it feasible and then, performing exchange moves to include in S_i elements that are not in S_g.

IV. Computational Results

Computational results compares the hybrid SO with PR algorithm, `GRASP+SO+PR` algorithm with Davoodi's algorithm `MOAkBCL`, and with the most competitive evolutionary algorithms (`GRASP+SO+PR`, `MOEA/D`, `NSGA-II` and `SPEA2`). We have considered the instances used in [1].

Table I shows the multi-objective metrics to compare the quality of the algorithms when solving the k-BLC$_{123}$ problem on the WorkSpace and Large-scale instances. Results of this table illustrate that our algorithm outperforms the other algorithms since the `SO+PR` gets considerably better results in all the multi-objective metrics. The only metric that is similar is the spread, showing that the dispersion of the efficient solutions obtained by all the algorithms is quite similar.

TABLE I
AVERAGE RESULTS FOR THE k-BLC$_{123}$ ON THE WORKSPACE AND LARGE-SCALE INSTANCES.

	$\|A\|$	$C(A,R)$	Δ	HV	ϵ	GD
`MOAkBCL`	6.00	0.78	0.98	0.26	1.31	133.43
`GRASP+SO+PR`	**10.88**	**0.17**	**0.96**	**0.40**	**0.06**	**105.56**
`MOEA/D`	5.38	0.76	0.97	0.28	0.48	151.42
`NSGA-II`	7.50	0.57	**0.96**	0.35	0.56	126.34
`SPEA2`	7.13	0.58	0.98	0.36	0.61	131.54

Next, the results obtained when solving the k-BLC$_{12}$ problem are included, also considering the instances used in [1]. Following the same reasoning, we have solved the same problem considering just one balancing function. To that end, Table II compares the quality of the algorithms when solving the k-BLC$_{12}$ problem on the S1 and A3 instances. The multi-objective metrics show that, on average, the `MOAkBCL` algorithm slightly outperforms the results obtained by the `GRASP+SO+PR` algorithm. In order to prove if the results obtained by the `MOAkBCL` and the `GRASP+SO+PR` algorithms are significant different, we have applied the Wilcoxon test. The p-values obtained are: 0.673, 0.799, 0.025, 0.612, 0.779 and 0.674 for test conducted on the number of efficient points obtained by each algorithm, the coverage, the spread, the hypervolume, the ϵ-constraint and the generational distance, respectively. As we can see, considering a p-value equal to 0.01, there is no significant difference in the results obtained by both algorithms, which statistically demonstrates that both algorithms have a similar performance.

TABLE II
AVERAGE RESULTS FOR THE k-BCL$_{12}$ ON THE S1 AND A3 INSTANCES.

	$\|A\|$	$C(A,R)$	Δ	HV	ϵ	GD
`MOAkBCL`	**5.63**	**0.26**	0.93	**0.28**	**0.80**	**177.73**
`GRASP+SO+PR`	5.38	0.33	**0.84**	0.17	0.86	184.15
`MOEA/D`	3.25	0.60	0.90	0.13	1.08	231.45
`NSGA-II`	3.13	0.74	0.90	0.12	1.11	229.44
`SPEA2`	2.25	0.72	**0.84**	0.13	1.04	254.66

References

[1] M. Davoodi, "k-Balanced Center Location problem: A new multi-objective facility location problem," Computers & Operations Research, vol. 105, pp. 68–84, 2019.

[2] J. Sánchez-Oro, A.D. López-Sánchez-Oro, A. Martínez-Gavara, A.G. Hernández-Díaz, and A.Duarte, "A Hybrid Strategic Oscillation with Path Relinking Algorithm for the Multiobjective k-Balanced Center Location Problem," Mathematics, vol. 9, pp. 853, 2021

Complete Shipment constraints in Container Loading Problem

I. Giménez-Palacios
Department of Mathematics
University of Castilla-La Mancha
Albacete, Spain
ivan.gimenez@uclm.es

M.T. Alonso
Department of Mathematics
University of Castilla-La Mancha
Albacete, Spain
mariateresa.alonso@uclm.es

R. Álvarez-Valdés
Department of Statistics and Operations Research
University of Valencia
Valencia, Spain
ramon.alvarez@uv.es

F. Parreño
Department of Mathematics
University of Castilla-La Mancha
Albacete, Spain
francisco.parreno@uclm.es

***Abstract*—In this study we focus on a special type of logistic constraint in Container Loading Problem (CLP): complete shipment constraints, which ensure that either all or none of the boxes in each customer order are loaded. Although these constraints have seldom been considered in the literature, they arise very often in practice. We develop four heuristic strategies to deal with them efficiently. An extensive computational study compares the relative performance of these strategies and shows that when a good strategy is used the impact of these constraints on the solution of the CLP is quite small. This work is completely based on *Logistic constraints in container loading problems: the impact of complete shipment conditions* (TOP, 2021, 29(1), pp. 177–203).**

***Index Terms*—Container Loading, Logistic Constraints, VNS, Integer Models**

I. Introduction

Efficient use of vehicles in transportation requires solving a packing problem to decide how to place boxes so to as maximize the total volume occupied. This problem is usually known as the CLP. The basic version of the CLP includes only geometric constraints. However, to be useful in practical applications, many other constraints have to be taken into account besides the basic geometric conditions. A list of these constraints was proposed by [1]. More recently, [2] revisited this list, proposing a classification of constraints and reviewing how they have been considered in the algorithms developed for the CLP. Practical constraints can be classified into two main groups, physical and logistic. Some logistic constraints, such as complete shipment constraints, have received very little attention, although there are many situations in which these constraints are imposed. In the problem that inspired this study, a large furniture factory sends its products to retail shops which in turn serve their customers. Very often, customer orders include large items that are packed into several boxes to be loaded onto trucks. Customers and shops do not want to receive only a subset of these boxes and keep them in storage until the remaining boxes arrive. This paper focuses on these complete shipment constraints and their effect on the solutions of the CLP. We develop four heuristic strategies to solve large instances.

II. A VNS algorithm for container loading

In several of the procedures developed in the following sections, we use the Variable Neighborhood Search (VNS) algorithm developed by [3] as a starting point, with some extensions to make it more efficient. The initial solution x is built by a constructive algorithm based on maximal-spaces. We select the maximal empty space nearest to a corner to be filled with boxes that we select taking as a criteria decision the volume occupied, the wasted space, the number of boxes and how they fit in the space. Boxes can be combined forming blocks of boxes, from different types and different orientations. Then, we define six neighborhoods that are used in the search: block reduction, column insertion, box insertion, emptying a region and refilling it using the constructive algorithm, emptying a region and refilling it selecting the block that best fits and removing a percentage of the last boxes loaded and refilling the space with the constructive algorithm.

III. Solution approaches with complete shipment

This section presents 4 different techniques developed for solving the CLP with complete shipment constraints heuristically.

A. A local a priori strategy: selecting the next order to load

A simple loading strategy is to sort the orders into a list according to some criterion, such as volume or number of boxes, take the next order on the list, and try to load all its boxes using the constructive deterministic algorithm which is part of the VNS procedure. If all of them are loaded, the next order is considered. If all the boxes cannot be loaded, the boxes in the incomplete order that have been loaded are removed and the process moves on to the next order. When all the orders on the list have been considered, some of them will have been loaded, resulting in a feasible solution.

B. A global a priori strategy: selecting a set of orders to load

Instead of selecting one order at a time to be loaded, an alternative consists in selecting a subset of orders to be loaded together. For that, we take as an upper bound the total volume we are able to load without complete shipment using the original VNS. With this value, TV_1, we solve a knapsack problem to obtain a set of orders that we try to load with the VNS. If we can load all the orders, the procedure ends. If not, we update the upper bound with the total volume obtained in the last VNS execution, and solve again the knapsack problem to obtain a different list. This procedure continues until we get a list of orders that we can load completely with the VNS. In this moment, we try to recover some of the volume that we wasted in the last update of the upper bound. That is, if in iteration k, with TV_k as an upper bound, we obtain a solution, then we try the same procedure but taking as an upper bound $(TVk - 1 + TV_k)/2$. The first time the boxes in the selected orders do not fit, the process ends with the last feasible solution obtained.

C. An a posteriori strategy

A different approach will be first to load as many boxes as possible into the container and then to assign the loaded boxes to orders in an a posteriori strategy. For the first step, we can use the VNS algorithm. Let n_j be the number of of type j in the solution of the VNS. The optimal assignment of boxes to orders can be obtained by solving an integer linear model. If we define variables z_i, with value 1 if order i is completely loaded, and 0 otherwise, and x_{ij} as the number of type j boxes assigned to order i, the model is:

$$Maximize \quad \sum_i V_i z_i \tag{1}$$

$$st: \quad \sum_i x_{ij} = n_j \qquad \forall j \tag{2}$$

$$x_{ij} \geq z_i n_{ij} \qquad \forall i, \forall j \tag{3}$$

This a posteriori assignment strategy does not guarantee a large number of complete orders by itself. In order to improve the solutions obtained by this strategy, we have developed a procedure that divides the set of incomplete orders in the solution into two sets S^C, the set of orders to be completed, and S^R, the set of orders whose boxes in the solution will be removed to make room for the boxes of orders in S^C. To determine S^C, we solve an integer problem to assign the unassigned boxes u_j, removing the orders that have been completed. With this model, we obtain S^C and S^R, so we remove orders from S^R and try to load the boxes to complete orders in S^C.

D. A mixed strategy

An alternative to the previous strategy is not to fill the container up completely but only to a given percentage of its volume. When this percentage is reached, the filling process stops and the model of expressions (1)-(3) is solved to assign the loaded boxes to orders. Orders i in the solution that are not complete ($z_i = 0$), but have some boxes in the solution ($\exists j \,|\, x_{ij} \geq 1$), are considered together for completion, using a deterministic constructive algorithm. If all these boxes are loaded, the remaining orders are considered for loading, one at a time, selected using the three criteria described in the previous strategy. If not all the boxes can be loaded, all orders, with or without boxes in the partial solution, are considered for loading, one at a time, using the same procedure.

IV. Computational study

TABLE I
Results of VNS without complete shipment and all four procedures

Instance	VNS	1st	2nd	3rd	4th
BR1	95,5	83,9	94,7	94,3	90,6
BR2	96,3	82,0	95,0	94,3	90,0
BR3	96,1	81,2	94,9	93,7	89,0
BR4	95,9	80,9	94,7	93,0	88,3
BR5	95,6	80,7	94,4	92,2	87,8
BR6	95,3	80,5	94,0	91,0	87,1
BR7	94,7	80,3	93,4	89,0	86,4
BR8	93,7	80,2	92,2	86,7	86,4
BR9	93,2	80,2	91,6	85,8	85,9
BR10	92,6	80,2	91,2	85,0	85,6
BR11	92,6	80,1	90,7	84,7	85,4
BR12	92,0	80,2	90,6	84,3	85,2
BR13	91,8	80,2	90,3	84,0	84,9
BR14	91,9	80,1	90,2	84,0	84,7
BR15	91,8	80,2	89,9	83,5	84,6
Average	93,9	80,7	92,5	88,4	86,8

In Table I we can see the summarized results of all the procedures we defined previously, and also of the original VNS without complete shipment. We see that the first procedure is the worst performing, as we expected since this procedure is the least elaborated. In the other three, we see a big improvement in comparison. However, in the more heterogeneous instances, the improvement over the first method decreases with the exception of procedure 2. As a conclusion, we can see that, with an adequate method such as procedure 2, the impact of the complete shipment is only 1.4% on average in volume of occupied space.

References

[1] Bischoff EE, Ratcliff MSW (1995) Issues in the development of approaches to container loading. Omega 23(4):377-390.
[2] Bortfeldt A, Wäscher G (2013) Constraints in container loading. A state of the art review. Eur J Oper Res 229(1):1-20.
[3] Parreño F, Alvarez-Valdes R, Oliveira JF, Tamarit JM (2010) Neighborhood structures for the container loading problem: a VNS implementation. J Heuristics 16:1-22.

Scatter search for the capacitated dispersion problem

Anna Martínez-Gavara
Dept. Estadística i Inv. Operativa
Universitat de València
València, Spain
gavara@uv.es

Jesús Sánchez-Oro
Dept. Ciencias de la Computación
Unirversidad Rey Juan Carlos
Madrid, Spain
jesus.sanchezoro@urjc.es

Rafael Martí
Dept. Estadística i Inv. Operativa
Universitat de València
València, Spain
Rafael.Marti@uv.es

***Abstract*—The capacitated dispersion problem (CDP) is an extension of the well-known p-dispersion problem. In this variant, each element has associated a storage capacity, so that CDP consists of selecting a subset of elements from a given set satisfying a capacity constraint in such a way that the minimum distance among all pair of elements is maximized. A mathematical model is proposed to optimally solve small and medium size instances. Finally, a heuristic based on scatter search is developed to obtain significant results compared with previous metaheuristics.**

***Index Terms*—metaheuristics, dispersion problems, combinatorial optimization**

I. Introduction

Diversity and dispersion maximization is a class of interesting and relevant combinatorial optimization problems both in terms of theoretical significance and practical applications [2], [4]. These problems have gained special attention in the last decade where different metaheuristics have been proposed to solve them [6], [8], [10], [14].

The most popular models in dispersion maximization are the Maximum Diversity Problem (MDP), and the Max-Min dispersion problem (MMDP), also known as p-dispersion problem. In the former model, the objective function is based on the sum of the distances, while in the latter the goal is to maximize the minimum distance between the pairs of selected points.

Recently, Rosenkrantz et al. [12] propose a more realistic variant of the MMDP in the context of location where the standard size constraint setting the number of elements is replaced by a capacity constraint. This new restriction ensures a minimum storage level B to provide a service. Given a graph $G = (V, E)$, let V be a finite collection of points (elements) with edge distances d_{ij} for every pair $i, j \in V$. Let define c_i as the capacity of element $i \in V$ and B the total capacity required. Let $\mathbb{P}$ be the set of all subsets P of V whose elements satisfy the capacity constraint, i.e.,

$$\mathbb{P} = \left\{ P \subset V : \sum_{i \in P} c_i \geq B \right\}.$$

This research has been partially supported by the Spanish Ministry with grant ref. PGC2018-0953322-B-C21 and PGC2018-0953322-B-C22 /MCIU/AEI/FEDER-UE.

Then, in a similar way than the p-dispersion problem, the capacitated dispersion problem (CDP) can be expressed as follows:

$$\text{(CDP): find } P^* = \arg\max_{P \in \mathbb{P}} \min_{i,j \in P} d_{ij}.$$

In [7], we deal with this new model. The goal is twofold:

- to improve the mathematical formulation to solve medium size instances on a small fraction of the time employed by the previous model.
- to propose a new heuristic method to deal with large instances (with up to 500 elements) that the improved model cannot solve.

The empirical analysis in [7] shows that the proposed strategies outperform in quality and time the previous approaches.

II. Mathematical model

In 2016, Sayyady & Fathi [13] propose an integer programming approach for the p-dispersion problem which is able to optimally solve large size instances. The main idea is to solve it as a sequence of node packing problems in which, given a threshold value l, the optimal solution (of the node packing problem) provides a subset of elements with minimum distance larger than or equal to $l \in [d_{min}, d_{max}]$. The method performs a binary search over the ordered set of different distances in the graph, based on the minimum distance between consecutive values, until a set of p elements is found. This solution turns out to be the optimal solution of the p-dispersion problem.

Specifically, let $G(l)$ be a graph with the set V of n elements of the original graph $G = (V, E)$, and the set of edges $E(l) = \{(i, j) \in E : d_{ij} < l\}$. Then the node packing problem consists of finding a maximum cardinality subset of elements so that no two elements in this subset are adjacent to each other. It can be formulated with binary variables, x_i, indicating if element i is selected as

$$\begin{aligned} \max \quad & \sum_{i \in V} x_i \\ \text{s. t.} \quad & x_i + x_j \leq 1 \quad \forall\ (i, j) \in E(l) \\ & x_i \in \{0, 1\} \quad \forall\ i \in V \end{aligned} \tag{1}$$

In [7], we adapt the node packing model to the CDP. The model is compared with the classical model used in [9] when solving the CDP on the 100 previously reported instances.

III. Solution approaches

The work of Rosenkrantz et al. [12] is mainly theoretical, however the authors also propose a simple heuristic `TI` for this problem based on a binary search over the distances. The authors prove that their procedure `TI` has performance guarantee of 2. In 2020, Peiró et al. [9] propose a metaheuristic that hybridize GRASP [11] with a Strategic Oscillation post-processing [1] to obtain high quality solutions for the CDP, called `SO`.

To solve CDP, we propose in [7] a metaheuristic based on the Scatter Search framework. This methodology consists of five methods [3]: Diversification generation (DG), Improvement , Reference set update (RefSet), Subset generation, and Combination. We have implemented two of them in their standard form, the RefSet and the subset generation. The main characteristics of the remaining three methods are described in [7]. In fact, two versions of SS are proposed. The first one, `SS1`, consists of the following four steps:

1) Generate $Psize$ solutions with DG.
2) Create population P with $Psize$ improved solutions.
3) Create the RefSet with the best b solutions in P:
   i. Select the best $b/2$ solutions in P in terms of quality.
   ii. Select the best $b/2$ solutions in P in terms of diversity.
4) Apply the combination method (based with the Path Relinking methodology) to each pair of solutions in the RefSet.

The parameter b is empirically adjusted in the numerical experiments and it refers to the size of selected solutions chosen from the set P of improved solutions with size $Psize$. `SS1` is designed to perform one iteration so that it would be competitive in running time.

In contrast to `SS1`, the second SS method, `SS2`, is competitive in quality and it requires longer CPU time than `SS1`. `SS2` starts with the same four steps of `SS1`, however it performs several iterations. The method ends after all pairs of solutions have been subjected to the combination method and none of the new solutions are admitted to RefSet.

IV. Numerical experiments

We perform two foremost experiments in [7] on a benchmark of 100 instances publicly available [5]. In the first one, we compare the standard mathematical model `CDP_standard` with our adaptation from the Node Packing Problem `CDP_NodePacking`. Gurobi is able to solve with the new formulation all the small and medium size instances (less than 150 elements) in one second, and some of 500 elements in a few seconds. However, there are 10 large instances for which there are a relatively large differences between the upper and lower bounds reported by Gurobi. To sum up, the new formulation clearly beats the standard model in quality and CPU time.

In the second main experiment, we compare our two new methods: `SS1` and `SS2` with the previous heuristics algorithms, `TI` and `SO`. As expected, small instances (with 50 elements) are relatively easy to solve with all the heuristic methods (with the exception of `TI`). On the contrary, large instances (with 500 elements) are difficult to solve by heuristics. In this cases, numerical results indicate that `SS1` and `SS2` outperforms the previous heuristics. Specifically, the average deviation in the whole set of instances is 21.5%, 13.4%, 1.6%, and 0.1% for `TI`, `SO`, `SS1`, and `SS2`, respectively.

V. Conclusions

Our experiments with this problem reveal two important facts. The first one is that an adequate mathematical formulation has a huge impact on the ability of solvers, such as Gurobi, to target practical instances. The second point is that search strategies in metaheuristics make a difference in their performance.

References

[1] Fred Glover and Jin-Kao Hao. The case for strategic oscillation. *Annals of Operations Research*, 183(1):163–173, 2011.

[2] Fred Glover, Ching-Chung Kuo, and Krishna S. Dhir. A discrete optimization model for preserving biological diversity. *Applied Mathematical Modelling*, 19(11):696–701, 1995.

[3] Fred Glover, Manuel Laguna, and Rafael Martí. Scatter search and path relinking: Advances and applications. In *Handbook of metaheuristics*, pages 1–35. Springer, 2003.

[4] Ching-Chung Kuo, Fred Glover, and Krishna S. Dhir. Analyzing and Modeling the Maximum Diversity Problem by Zero-One Programming. *Decision Sciences*, 24(6):1171–1185, 11 1993.

[5] Rafael Martí, Abraham Duarte, Anna Martínez-Gavara, and Jesús Sánchez-Oro. The MDPLIB 2.0 library of benchmark instances for diversity problems, university of valencia. https://www.uv.es/rmarti/paper/mdp.html, 2021.

[6] Rafael Martí, Micael Gallego, Abraham Duarte, and Eduardo G Pardo. Heuristics and metaheuristics for the maximum diversity problem. *Journal of Heuristics*, 19(4):591–615, 2013.

[7] Rafael Martí, Anna Martínez-Gavara, and Jesús Sánchez-Oro. The capacitated dispersion problem: an optimization model and a memetic algorithm. *Memetic Computing*, 13:131–146, 2021.

[8] Rafael Martí and Fernando Sandoya. GRASP and path relinking for the equitable dispersion problem. *Computers & Operations Research*, 40(12):3091–3099, 12 2013.

[9] Juanjo Peiró, Iris Jiménez, José Laguardia, and Rafael Martí. Heuristics for the capacitated dispersion problem. *International transactions in operational research*, 28(1):119–141, 2021.

[10] Daniel Cosmin Porumbel, Jin Kao Hao, and Fred Glover. A simple and effective algorithm for the MaxMin diversity problem. *Annals of Operations Research*, 186(1):275–293, 6 2011.

[11] Mauricio GC Resende, Rafael Martí, Micael Gallego, and Abraham Duarte. GRASP and path relinking for the max–min diversity problem. *Computers & Operations Research*, 37(3):498–508, 2010.

[12] Daniel J. Rosenkrantz, Giri K. Tayi, and S. S. Ravi. Facility Dispersion Problems under Capacity and Cost Constraints. *Journal of Combinatorial Optimization*, 4(1):7–33, 2000.

[13] Fatemeh Sayyady and Yahya Fathi. An integer programming approach for solving the p-dispersion problem. *European Journal of Operational Research*, 253(1):216–225, 8 2016.

[14] Qinghua Wu and Jin-Kao Hao. A hybrid metaheuristic method for the Maximum Diversity Problem. *European Journal of Operational Research*, 231(2):452–464, 12 2013.

A review and empirical analysis of diversity and dispersion problems

Rafael Martí
Dept. Estadística i Inv. Operativa
Universitat de València
València, Spain
Rafael.Marti@uv.es

Anna Martínez-Gavara
Dept. Estadística i Inv. Operativa
Universitat de València
València, Spain
gavara@uv.es

Segio Pérez-Peló
Dept. Ciencias de la Computación
Unirversidad Rey Juan Carlos
Madrid, Spain
sergio.perez.pelo@urjc.es

Jesús Sánchez-Oro
Dept. Ciencias de la Computación
Unirversidad Rey Juan Carlos
Madrid, Spain
jesus.sanchezoro@urjc.es

***Abstract*—The diversity or dispersion problem requires the selection of a specific number of elements from a given set, in such a way that the dissimilarity among the selected elements is maximized. This problem comprises a variety of models that have received a lot of attention and have been solved with many complex heuristics in the last 30 years. In this work, we review the different mathematical models and discuss the metaheuristic methodologies proposed to deal with them.**

***Index Terms*—metaheuristics, dispersion problems, combinatorial optimization**

I. Introduction

Diversity or dispersion maximization consists in selecting a subset of elements from a given set in such a way that the distance among them is maximized. There has been a growing interest in these problems in the last 30 years, and several models have been proposed to reflect the notion of diversity and dispersion depending on the specific practical application. These models have applications in workforce management, ecological preservation or location problem, among others [3]. The literature in diversity problems, closely linked to location models, is vast and includes many solution methodologies, from simple heuristics to complex metaheuristics. These problems are still developing and exhibit a huge potential in new areas, such as those including side constraints, that deserve the attention of researchers in optimization.

In [8], a comprehensive survey of the different mathematical models is presented. We review the metaheuristics methodologies proposed to solve them. The work includes a critical analysis from an Operation Research (OR) perspective of the previous developments, which permits us to establish the most appropriate models, their connection with practical problems in terms of dispersion and representativeness, and the open problems that are still a challenge.

This research has been partially supported by the Spanish Ministry with grant ref. PGC2018-0953322-B-C21 and PGC2018-0953322-B-C22 /MCIU/AEI/FEDER-UE.

II. Chronological analysis

Starting when the first models were proposed, we identify three periods of time. The **early period**, from 1977 to 2000, is where the first models (MaxMin and MaxSum [2], [4]) appeared. In the OR area, only few papers addressed these problems in this period, and solved them with relatively simple algorithms. However, in other fields of science, such as sociology or biology, diversity maximization received much more attention.

The first metaheuristics to deal with large instances were developed in the second period, called **expansion period**, covering from 2000 to 2010. Special attention deserves the works of Duarte et al. [1] and Prokopyev et al. [10]. In the former paper, the authors proposed a Tabu Search and GRASP methodologies to solve the MaxSum model inviting the metaheuristic community to deal with this family of problems. In the latter paper, different measures are proposed to model diversity maximization, so that new dispersion models are introduced: the MaxMinSum, the MaxMean, and the MinDiff. This period ends with the empirical analysis of the most efficient methods at that time for some models (see the work of [7]). The boundaries defining the area of maximizing diversity were expanded with the inclusion of more realistic models built with capacity and cost constraints.

We identify the last period as the **development period**, which started in 2011. In this period complex metaheuristics are developed in order to improve in quality and time the efficient methods published in the previous period. Regarding exact methods, new efficient integer formulations are proposed by [13] and [12] for the MaxMin model. These new methods are able to solve large size problems.

In the previous period, Rosenkrantz [11] proposed an extension of the dispersion problem by including capacity and cost constraints in the classic diversity models. In [9], the authors adapted both Tabu Search and GRASP methodologies to these theoretical models, opening new research opportunities and

modeling a wide range of real problems.

III. Mathematical Models

To the best of our knowledge, Kuby [4] published the first discrete versions of diversity problems where he proposes a linear integer formulation for the MaxSum and MaxMin Diversity problems, MDP and MMDP respectively. The latter is also known as the p-dispersion problem.

Given a graph $G = (V, E)$, let V be a finite collection of n elements and E is the set of edges. Let d_{ij} be the dissimilarity measure between elements i and j in V. Let $\mathbb{P}$ be the set of all subsets of p elements of V, i.e.,

$$\mathbb{P} = \{P \subset V : |P| = p\}.$$

Then, MDP can be expressed as

$$\text{(MDP): find } P^* = \arg\max_{P \in \mathbb{P}} \sum_{i,j \in P} d_{ij}.$$

Then, in a similar way, the p-dispersion problem can be stated as follows:

$$\text{(MMDP): find } P^* = \arg\max_{P \in \mathbb{P}} \min_{i,j \in P} d_{ij}.$$

Since then, new diversity models have been proposed. On one hand, Prokopyev et al. [10] introduce new ways of measuring diversity. On the other, new models come from limitations on storage capacity and a maximum budget [11]. Table I summarizes the most relevant diversity models introduced so far.

TABLE I
Diversity models

Problem	Obj. function	First ref.
MaxSum	$\sum_{i<j,i,j \in P} d_{ij}$	[4]
MaxMin	$\min_{i<j,i,j \in P} d_{ij}$	[4]
MaxMin/Cap/Cost	$\min_{i<j,i,j \in P} d_{ij}$	[11]
Max-Mean	$\frac{\sum_{i<j,i,j \in P} d_{ij}}{\lvert P \rvert}$	[10]
Max-MinSum	$\min_{i \in P} \sum_{j \in P, j \neq i} d_{ij}$	[10]
Min-Diff	$\max_{i \in P} \sum_{j \in P, j \neq i} d_{ij} - \min_{i \in P} \sum_{j \in P, j \neq i} d_{ij}$	[10]

IV. Computational experiments

Most of the published works in diversity and dispersion problems propose solution approaches based on metaheuristics. In [8], we present an empirical analysis performed executing some of the best methods developed to solve some of the models and discuss the current solving strategies to clearly identify the state-of-the art methods for the main diversity models.

We also revise the well-established benchmark set of instances, the MDPLIB [5], that has been employed widely to evaluate heuristics for all dispersion models. Some type of instances in [5] are not well suited for some of models, and others are too simple for nowadays complex methods. We removed some small instances, and added new ones obtaining a new library with 770 instances. We call this new library, MDPLIB 2.0 [6].

V. Conclusions

In this review, we analyze the different models and solving methods to deal with diversity problems. Although it has been the subject of intense study in the last 20 years, there is still much room for improvement, especially in the constrained models recently considered.

Acknowledgment

We would like to thank the authors who kindly made their codes available to us and all the researchers who contributed to diversity problems. Additionally, we would like to thank to some friends and colleagues that significantly contributed to the field: Mike Kuby; Fred Glover and Manuel Laguna; Abraham Duarte and Francisco Parreño; Mauricio Resende, Simone Martins, Luiz Ochi, and Geiza Silva; Xiangjin Lai, Zhipeng Lü, and Jin-Kao Hao; Roberto Aringhieri and Roberto Cordone; Oleg Prokopyev; Jack Brimberg, Nenad Mladenović, Raca Todosijević, and Dragan Urošević.

References

[1] Abraham Duarte and Rafael Martí. Tabu search and GRASP for the maximum diversity problem. *European Journal of Operational Research*, 178(1):71–84, 4 2007.

[2] Erhan Erkut. The discrete p-dispersion problem. *European Journal of Operational Research*, 46(1):48–60, 1990.

[3] Fred Glover, Ching-Chung Kuo, and Krishna S. Dhir. Heuristic algorithms for the maximum diversity problem. *Journal of Information and Optimization Sciences*, 19(1):109–132, 1 1998.

[4] Michael J. Kuby. Programming models for facility dispersion: the p-dispersion and maxisum dispersion problems. *Mathematical and Computer Modelling*, 10(10):792, 1988.

[5] Rafael Martí and Abraham Duarte. MDPLIB - maximum diversity problem library. https://www.uv.es/rmarti/paper/mdp.html, 2010.

[6] Rafael Martí, Abraham Duarte, Anna Martínez-Gavara, and Jesús Sánchez-Oro. The MDPLIB 2.0 library of benchmark instances for diversity problems, university of valencia. https://www.uv.es/rmarti/paper/mdp.html, 2021.

[7] Rafael Martí, Micael Gallego, Abraham Duarte, and Eduardo G Pardo. Heuristics and metaheuristics for the maximum diversity problem. *Journal of Heuristics*, 19(4):591–615, 2013.

[8] Rafael Martí, Anna Martínez-Gavara, Sergio Pérez-Peló, and Jesús Sánchez-Oro. Discrete diversity and dispersion maximization. a review and an empirical analysis from an or perspective. *European Journal of Operation Research*, 2021.

[9] Anna Martínez-Gavara, Teresa Corberán, and Rafael Martí. GRASP and Tabu Search for the Generalized Dispersion Problem. *Expert Systems with Applications*, 173:114703, 2021.

[10] Oleg A. Prokopyev, Nan Kong, and Dayna L. Martinez-Torres. The equitable dispersion problem. *European Journal of Operational Research*, 197(1):59–67, 2009.

[11] Daniel J. Rosenkrantz, Giri K. Tayi, and S. S. Ravi. Facility Dispersion Problems under Capacity and Cost Constraints. *Journal of Combinatorial Optimization*, 4(1):7–33, 2000.

[12] David Sayah and Stefan Irnich. A new compact formulation for the discrete p-dispersion problem. *European Journal of Operational Research*, 256(1):62–67, 1 2017.

[13] Fatemeh Sayyady and Yahya Fathi. An integer programming approach for solving the p-dispersion problem. *European Journal of Operational Research*, 253(1):216–225, 8 2016.

X Simposio de Teoría y Aplicaciones de Minería de Datos



X Simposio de Teoría y Aplicaciones de Minería de Datos

Sesión General



Nearest Neighbors-based Forecasting for Electricity Demand Time Series in Streaming*

*Note: The full contents of this paper have been published in the volume Lecture Notes in Artificial Intelligence 12882 (LNAI 12882)

L. Melgar-García
Data Science & Big Data Lab
Pablo de Olavide University
Seville, Spain
lmelgar@upo.es

D. Gutiérrez-Avilés
Department of Computer Science
University of Seville
Seville, Spain
dgutierrez3@us.es

C. Rubio-Escudero
Department of Computer Science
University of Seville
Seville, Spain
crubioescudero@us.es

A. Troncoso
Data Science & Big Data Lab
Pablo de Olavide University
Seville, Spain
atrolor@upo.es

***Abstract*—This paper presents a new forecasting algorithm for time series in streaming named StreamWNN. The methodology has two well-differentiated stages: the algorithm searches for the nearest neighbors to generate an initial prediction model in the batch phase. Then, an online phase is carried out when the time series arrives in streaming. In particular, the nearest neighbor of the streaming data from the training set is computed and the nearest neighbors, previously computed in the batch phase, of this nearest neighbor are used to obtain the predictions. Results using the electricity consumption time series are reported, showing a remarkable performance of the proposed algorithm in terms of forecasting errors when compared to a nearest neighbors-based benchmark algorithm. The running times for the predictions are also remarkable.**

***Index Terms*—Forecasting, nearest neighbors, streaming time series, electricity demand.**

DOI: https://doi.org/10.1007/978-3-030-85713-4_18

STree: a Single Multi-class Oblique Decision Tree Based on Support Vector Machines*

*Note: The full contents of this paper have been published in the volume Lecture Notes in Artificial Intelligence 12882 (LNAI 12882)

Ricardo Montañana
Computing Systems Department
University of Castilla-La Mancha
Albacete, Spain
Ricardo.Montanana@alu.uclm.es

José A. Gámez
Computing Systems Department
University of Castilla-La Mancha
Albacete, Spain
Jose.Gamez@uclm.es

José M. Puerta
Computing Systems Department
University of Castilla-La Mancha
Albacete, Spain
Jose.Puerta@uclm.es

***Abstract*—We propose a new oblique decision tree algorithm based on support vector machines. Our algorithm produces a single model for a multi-class target variable. On the contrary to previous works that manage the multi-class problem by using clustering at each split, we test all the one-vs-rest labels at each split, choosing the one which minimizes an impurity measure. The experimental evaluation carried out over 49 datasets shows that our algorithm is ranked before those used for comparison, and significantly outperforms all of them when the SVM hyper-parameters are carefully tuned.**

***Index Terms*—Oblique decision trees, Supervised Classification, SVM.**

DOI: https://doi.org/10.1007/978-3-030-85713-4_6

Studying the Effect of Different L_p Norms in the Context of Time Series Ordinal Classification*

*Note: The full contents of this paper have been published in the volume Lecture Notes in Artificial Intelligence 12882 (LNAI 12882)

David Guijo-Rubio
Dept. of Computer Science and Numerical Analysis
University of Córdoba
Córdoba, Spain
dguijo@uco.es

Víctor Manuel Vargas
Dept. of Computer Science and Numerical Analysis
University of Córdoba
Córdoba, Spain
vvargas@uco.es

Pedro Antonio Gutiérrez
Dept. of Computer Science and Numerical Analysis
University of Córdoba
Córdoba, Spain
pagutierrez@uco.es

César Hervás-Martínez
Data Science & Big Data Lab
Pablo de Olavide University
Seville, Spain
chervas@uco.es

***Abstract*—Time Series Ordinal Classification (TSOC) is yet an unexplored field of machine learning consisting in the classification of time series whose labels follow a natural order relationship between them. In this context, a well-known approach for time series nominal classification was previously used: the Shapelet Transform (ST). The exploitation of the ordinal information was included in two steps of the ST algorithm: 1) by using the Pearson's determination coefficient (R^2) for computing the quality of the shapelets, which favours shapelets with better ordering, and 2) by applying an ordinal classifier instead of a nominal one to the transformed dataset. For this, the distance between labels was represented by the absolute value of the difference between the corresponding ranks, i.e. by the L_1 norm. In this paper, we study the behaviour of different L_p norms for representing class distances in ordinal regression, evaluating 9 different L_p norms with 7 ordinal time series datasets from the UEA-UCR time series classification repository and 10 different ordinal classifiers. The results achieved demonstrate that the Pearson's determination coefficient using the $L_{1.9}$ norm in the computation of the difference between the shapelet and the time series labels achieves a significantly better performance when compared to the rest of the approaches, in terms of both Correct Classification Rate (CCR) and Average Mean Absolute Error (AMAE).**

***Index Terms*—Time Series, Ordinal Classification, L_p norm, Shapelet Quality Measures.**

DOI: https://doi.org/10.1007/978-3-030-85713-4_5

Universal Adversarial Examples in Speech Command Classification

Jon Vadillo
University of the Basque Country UPV/EHU
Donostia - San Sebastian, Spain
jon.vadillo@ehu.eus

Roberto Santana
University of the Basque Country UPV/EHU
Donostia - San Sebastian, Spain
roberto.santana@ehu.eus

***Abstract*—Adversarial examples are inputs intentionally perturbed with the aim of forcing a machine learning model to produce a wrong prediction, while the changes are not easily detectable by a human. Although this topic has been intensively studied in the image domain, classification tasks in the audio domain have received less attention. In this paper we generate universal perturbations for speech command classification, which are able to generalize across different models to a significant extent. Additionally, a novel analytical framework is proposed for the evaluation of universal perturbations under different levels of universality, demonstrating that the feasibility of generating effective perturbations decreases as the universality level increases. Finally, we propose the use of a more rigorous approach to measure the amount of distortion introduced by the adversarial perturbations, demonstrating that the methods employed by convention are not realistic in speech-based problems.**

***Index Terms*—Adversarial Examples, Deep Neural Networks, Speech Command Classification, Robust Speech Recognition**

I. Introduction

In [1], Szegedy et al. discovered that an attacker can fool a Deep Neural Network (DNN) with inputs which are slightly but adversarially modified: adversarial examples. These inputs are perturbed in such a way that the changes can not be detected by a human, while forcing a target model to misclassify or predict a specific output with high probability. The existence of such attacks questioned the reliability of DNNs in adversarial scenarios, and the reason for these vulnerabilities is still an open question [2]. The study of adversarial examples has focused primarily on the image domain and computer vision tasks [3], whereas domains such as text or audio have received much less attention. Bearing in mind that the properties of adversarial examples are not well understood yet, studying the effect of these attacks in domains other than image-based tasks can provide very useful insights about the general properties of adversarial examples.

In this paper, we study the feasibility of generating universal adversarial perturbations for the task of speech command classification. In particular, we evaluate how the feasibility of generating such perturbations changes when we consider different levels of universality, in terms of the number of classes for which the perturbation is able to fool the target model when the instances of such classes are perturbed. For this purpose, we introduce the taxonomy of $N-$class universality as a novel analytical framework for studying the existence of universal adversarial perturbations, and, based on this framework, we create different types of universal perturbations in the speech command classification tasks. Finally, we show that the distortion metrics used by convention to measure the amount of perturbation in audio adversarial examples are not realistic for speech related tasks, and address this limitation by proposing a more specific and rigorous evaluation framework. To facilitate future research, we have made our code publicly available, along with the models and dataset used in this paper. [1]

This work is supported by the Basque Government (ELKARTEK program, IT1244-19, and PRE_2019_1_0128 predoctoral grant), by the Spanish Ministry of Economy and Competitiveness MINECO (projects TIN2016-78365-R and PID2019-104966GB-I00) and by the Spanish Ministry of Science, Innovation and Universities (FPU19/03231 predoctoral grant).

II. Notation and assumptions

In this section we introduce the notation and present a number of assumptions used to derive our results. First, we address adversarial perturbations in the audio domain for the particular problem of voice command classification. Specifically, the task of classifying an audio waveform among a finite set of possible (and previously defined) labels. We will assume audio waveforms of fixed lengths and composed of values in a bounded range. We will define a classification model f as a function $f : X \to Y$, where $X \subseteq \mathbb{R}^d$ represents the d dimensional input space of audio waveforms, and Y the set of possible output labels, where $y_i \in \{y_1 \dots y_k\}$ represents the i-th label (commonly words or commands). We will assume an end-to-end differentiable classification process, from which all the information will be available, so that the gradients of the model's output can be computed with respect to the input audio waveform.

A. Adversarial examples

An adversarial example x' is defined as $x' = x + v$, where x represents the original (clean) input, and $v \in \mathbb{R}^d$ represents the adversarial perturbation. We assume that the original input x must be a sample correctly classified by the model, and $f(x') \neq f(x)$. In addition, the perturbation will be measured by a distortion metric $\varphi(x, x')$, $\varphi : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$, and will be restricted by a maximum threshold ϵ, so that $\varphi(x, x') \leq \epsilon$, to ensure that x' is as similar as possible to x.

[1] https://github.com/vadel/AudioUniversalPerturbations

Given a clean input sample x and a target class $y_t \neq f(x)$, a *targeted adversarial example* consists of a perturbed sample $x' = x + v$ which satisfies $f(x') = y_t$. In contrast, *untargeted adversarial examples* only require the output label to not be the same as the original one, that is, $f(x') \neq f(x)$.

Individual adversarial perturbations are unique perturbations crafted specifically to be applied to one particular input sample, so the same perturbation is not expected to work with a different input. In contrast, *universal adversarial perturbations* are input-agnostic perturbations which are able to fool the model independently of the input. In this work, we will consider three different types of universal adversarial perturbations, according to the number of classes for which they are expected to work:

- Level 1 - *Single-class* universal perturbations: perturbations expected to fool the target model only for inputs of one particular class.
- Level 2 - *Multiple-class* universal perturbations: perturbations expected to fool target model only for a particular subset of N classes, with $1 < N < k$.
- Level 3 - *Fully* universal perturbations: perturbations expected to fool a target model for every possible input, with complete independence of the original class.

Notice that the three levels are particular cases of an abstract type of N-class adversarial perturbations, where $1 \leq N \leq k$. However, for the sake of clarity we use this separation in levels to develop our experiments and analysis.

III. Related work

Universal (input-agnostic) adversarial perturbations were first reported in [4], for the task of image classification, in which the universal perturbation v is crafted by sampling a subset of input instances X, iteratively computing the minimal local perturbation $\triangle v_i$ that sends each instance $x_i \in X$ to the decision boundary of the model, and then adding this local perturbation to v. The computation of $\triangle v_i$ is performed using the DeepFool algorithm [5], an individual adversarial attack. In order to ensure that the norm of the perturbation is restricted $||v||_p \leq \xi$, after each modification, v is projected on the ℓ_p ball of radius ξ and centered at 0, using the following projection operator: $\mathcal{P}_{p,\xi}(v) = \arg\min_{v'} ||v - v'||_2$ s.t. $||v'||_p \leq \xi$.

The existence of universal adversarial perturbations in the audio domain was first addressed for the task of speech transcription [6], in which the approach proposed in [4] was employed, reformulated for that particular task. In concurrent works [7]–[10], universal perturbations have been generated for different audio classification tasks, such as speech classification, environmental sound classification or speaker recognition. In [8] a generative neural network is trained to generate universal perturbations. In [9], very short (subsecond) perturbations are introduced, which mimick environmental sounds to make them inconspicuous to humans. Closer to our work, in [7] the method proposed in [4] was also employed, however, replacing DeepFool by the Decoupled Direction and Norm attack [11], whereas we maintain the original algorithm, and introduce the required modifications for the method to work in the audio domain (see Section V). The authors also introduced a *penalty* method in which a gradient descent approach is performed to minimize the distortion and to maximize the attack strength, using a penalty coefficient to control the trade-off between those objectives.

In comparison to previous works, which address only *Level-3* universal perturbations, we study in depth the feasibility of generating perturbations with different levels of universality. Moreover, we propose more rigorous standards to measure the amount of distortion introduced by the perturbations in voice signals, showing that different distortion metrics should be applied in different parts of the signals to achieve representative results, and demonstrating that the approaches employed by convention in previous works are not realistic when applied to speech related problems.

IV. Speech Command Classification

In order to introduce and validate the proposed approaches, we will use the Speech Command Dataset [12], which consists of a set of WAV audio files of 30 different spoken commands. The duration of all the files is fixed to 1 second, and the sample-rate is 16kHz in all the samples, so that each audio waveform is composed of 16000 values, in the range $[-2^{15}, 2^{15}]$. The original dataset contains samples of 30 different labels. We will use a subset of ten classes, the standard labels selected in previous publications [12], [13]: "Yes", "No", "Up", "Down", "Left", "Right", "On", "Off", "Stop", and "Go". In addition to this set, we will consider two special classes: "Unknown" (a command different to the ones specified before) and "Silence" (no speech detected). As in previous works in the area [13]–[15], the target model that will be used to generate the adversarial examples is based on the CNN architecture described in [16]. Notice that the choice of the model is independent of the algorithm we have introduced. This model takes as input a single WAV audio file (with a sample rate of 16kHz) and transforms the signal from the time domain to the frequency domain, from which a set of MFCC features are extracted for different time intervals. This results in a two dimensional channel, which is fed into the following topology: two convolutional layers with a rectified linear unit (ReLU) activation function, a fully-connected layer and a final softmax layer. The implementation has been built on the code provided in [17] and [18], available from [2] and [3] respectively. The model has been trained using the second version of the Speech Command Dataset [12], achieving a Top-One Accuracy of 85.5% in the respective test dataset[4].

To analyze the effectiveness of the adversarial perturbations in a model other than the one used to generate them, we will use the pretrained speech recognition model available at [5]. Although this model has the same architecture as the one used to generate the attacks, it has different parameters and

[2] https://github.com/carlini/audio_adversarial_examples

[3] https://github.com/tensorflow/tensorflow/tree/master/tensorflow/examples/speech_commands

[4] download.tensorflow.org/data/speech_commands_test_set_v0.02.tar.gz

[5] download.tensorflow.org/models/speech_commands_v0.02.zip

is based on a different computation of the MFCC features. This model achieves a Top-One accuracy of 82.5% on the same test dataset as Target Model A. We verified that there exist important differences between the predictions given by the models, as shown in Table I, in which the initial accuracy of each model is computed per class.

TABLE I
INITIAL ACCURACY PER CLASS OF THE TARGET MODELS

Class	Target Model A	Target Model B
Silence	99.51	56.61
Unknown	66.42	71.08
Yes	94.03	91.17
No	74.57	82.47
Up	92.00	91.76
Down	80.79	79.31
Left	89.81	93.45
Right	88.64	90.66
On	87.12	87.63
Off	81.59	82.09
Stop	93.67	90.99
Go	77.36	72.14

V. METHODOLOGY: HILL CLIMBING REFORMULATION OF UNIVERSAL ADVERSARIAL PERTURBATIONS

To generate universal perturbations, we use Algorithm 1 (hereinafter referred to as UAP-HC), which is a reformulation of the UAP algorithm proposed for images [4]. The introduced changes were motivated because, after an initial experimentation with the original algorithm, the obtained results revealed that updating the universal perturbation v with every new local perturbation $\triangle v_i$ (assuming that $f(x_i) = f(x_i + v) \neq f(x_i + v + \triangle v_i)$, being x_i the input sample for which $\triangle v_i$ is crafted) led to a highly unstable evolution of the fooling rate, producing ineffective perturbations. To overcome this limitation, we employed a different acceptance criterion in the algorithm: the perturbation is updated in a *Hill-Climbing* fashion only if the update improves the global fooling rate of v, which is determined by the ratio of fooled samples of X when applying a perturbation (line 10). We verified that this reformulation was capable of significantly increasing the effectiveness of the perturbations in comparison to the original algorithm It is an open question whether the change of the acceptance criterion could lead to improvements in other audio tasks, since in the research reported in [6] and [7] the update rule is maintained without changes.

Note that this reformulation supposes a higher time complexity,[6] as we need to classify all the instances after every modification of the attack in order to evaluate the global fooling rate. However, the evaluation of the whole set is fully parallelizable, and, therefore, the computational time can be easily reduced in practice, with a negligible difference with respect to the original version of the algorithm, in which a sequential processing is required for all the remaining steps.

Apart from that, updating the perturbation only when the global fooling rate is improved may suppose a faster convergence than the original method, which can reduce the number of steps required to find quality solutions and counteract the increase in the time complexity. On the other hand, it may converge easier to local optima, leading to non-optimal solutions.

A. Experimental details

Regarding the parameters of the algorithm, the universal perturbation is bounded by the ℓ_2 norm, with a threshold value of $\xi = 0.1$. In addition, the maximum number of iterations allowed for the DeepFool algorithm is set to 100. The overshoot parameter of the DeepFool algorithm is set to 0.1. Finally, the fooling rate threshold of UAP-HC is set to 0.9 ($\alpha = 0.1$), and the algorithm is restricted to 5 *epochs*, that is, 5 complete passes through the entire set X.

The effectiveness of the perturbation in both the subset of samples used to craft the perturbation (*training set*) and the subset used to measure its effectiveness on unseen samples (*validation set*) is computed as the percentage of originally correctly classified samples for which the prediction of the model changes after the adversarial attack. We will refer to this metric as *fooling ratio* (*FR*) [4].

In all the experiments, the set used to craft the perturbations is composed of N samples randomly selected from the same training dataset used to train our target model (without replacement). In all the cases, 5 different trials are made, using different samples of the training set, and the results are reported for the perturbation that maximizes the FR on the training set. The validation set is the one used to test the initial accuracies of the target models in Section IV.

Finally, following the approach employed in previous works on adversarial examples in speech related tasks [6]–[8], [17], the relative loudness of the adversarial perturbation v with respect to an audio sample x is measured according to the following metrics, in Decibels (dB): $dB_{x,max}(v) = dB(v) - dB(x)$, where $dB(x) = \max_i \ 20 \cdot log_{10}(|x_i|)$. As the loudness of the perturbation should be lower than the loudness of the original signal, the difference is expected to be below zero. A distortion level of approximately -32dB is established as the maximum acceptable distortion, which roughly corresponds to the difference between ambient noise in a quiet room and a person talking [17], [19]. In our results, we report the mean relative loudness on the validation set to measure the distortion produced by the universal perturbation generated.

VI. RESULTS

A. Level 1 perturbations

We initially focused on the lowest universality level according to the taxonomy introduced, in which we attempt to generate a single perturbation which is able to fool any input of one particular class. This analysis may highlight the robustness or weakness of different classes against universal perturbations. For this level, a size of 1000 instances was used for the training set.

The FR in both training and validation sets and the mean distortion level of the perturbations are presented in Table II

[6]In particular, with respect to the original method, the introduced reformulation adds a term of $O(|X|^2 g_f(x))$ per epoch, being $g_f(x)$ the time complexity of making a single evaluation of the classifier $f(x)$.

Algorithm 1 UAP-HC

Input: A classification model f, a set of input samples X, a projection operator $\mathcal{P}_{p,\xi}$ and a fooling rate threshold α
Output: A universal perturbation v

$v \leftarrow$ initialize with zeros;
$foolingRate \leftarrow 0$;
while $foolingRate < 1 - \alpha$ **do**
  for $x_i \in X$ **do**
    Check that x_i is not already fooled by v:
    if $f(x_i) = f(x_i + v)$ **then**
      $\triangle v_i \leftarrow$ DeepFool$(x_i + v)$; Local perturbation
      Update v only if adding $\triangle v_i$ increases the global FR:
      $v' \leftarrow \mathcal{P}_{p,\xi}(v + \triangle v_i)$;
      $foolingRate' \leftarrow \mathbb{P}_{x \in X}(\ f(x) \neq f(x + v')\ )$;
      if $foolingRate < foolingRate'$ **then**
        $v \leftarrow v'$;
        $foolingRate \leftarrow foolingRate'$;
      end if
    end if
  end for
end while

for each class. According to the results, the effectiveness of the algorithm is remarkably high, also requiring a considerably low amount of distortion, far below the threshold of -32dB except for class *silence*.[7] For 7 out of 12 classes the best perturbation fools at least 70% of the samples in the validation set, and for 10 out of 12 classes it fools at least 60%. Clearly, generating universal perturbations is more difficult in some classes than in others, since the fooling ratios obtained are of a considerably different magnitude. Although some of the classes obtain fooling ratios above 80%, as is the case of *no*, others such as *silence*, *left* or *stop* are below 62%. In particular, in the case of the class *silence*, the FR is below 25% in both training and validation sets.

B. Level 2 perturbations

As explained in Section II, Level 2 universal perturbations attempt to create a single perturbation which is able to fool all the inputs of a particular subset of classes. Their study allows us to investigate how the difficulty of the problem changes among different subsets of classes, helping to identify those more vulnerable to universal perturbations. It also opens the question of whether Level 2 perturbations are less or more effective than Level 3 perturbations. Such insights could be used in future research as a basis to study different properties of universal attacks and to obtain a better understanding of these vulnerabilities.

The experimentation initially focuses on subsets of two classes. For illustration purposes, we selected a reduced and diverse set of pairs in terms of the phonetic similarity. The following pairs are considered: *no-go*, *up-off*, *yes-left*, *unknown-stop*, *silence-unknown*, and *no-up*. The training set is composed of 500 samples per class, forming a total of 1000 training samples.

[7]This effect on class *silence* is due to the fact that, due to the nature of the samples corresponding to that class, their loudness level is lower than for the rest of classes.

TABLE II
RESULTS OBTAINED IN LEVEL 1 UNIVERSAL PERTURBATIONS

Class	Target Model A		Target Model B		Mean dB
	Train	Valid	Train	Valid	
Sil.	23.80	19.46	12.96	12.55	-27.26
Unk.	72.70	73.06	21.30	31.38	-39.64
Yes	74.50	74.36	62.58	67.02	-41.72
No	86.50	83.77	4.38	7.19	-45.04
Up	84.20	75.45	9.17	7.95	-49.15
Down	71.50	65.55	14.15	12.73	-41.07
Left	52.30	49.73	13.40	17.66	-38.43
Right	68.70	63.82	12.29	15.04	-43.48
On	76.00	75.65	7.55	9.22	-42.75
Off	80.10	73.48	26.81	37.58	-43.73
Stop	61.40	61.82	19.77	25.40	-44.65
Go	87.80	80.06	29.78	32.41	-42.62

Results are shown in Table III. For each pair of classes, the FR has been computed considering the audio samples of the two classes involved (*Total*) as well as individually in each class (*C1* and *C2*). The mean distortion levels of the perturbations are also displayed in the table, computed from the validation set. According to the results, for the pairs *no-go*, *up-off* and *no-up* we obtained remarkably high fooling ratios, being also the most balanced pairs regarding the values obtained at each class. On the contrary, for the remaining pairs we obtained a final FR below 50%, being also highly unbalanced on the ratios per class, with the exception of the pair *unknown-stop*, in which the difference between both classes is approximately 13% in the training set and approximately 5% in the validation set.

C. Level 3 perturbations

Level 3 universal perturbations comprise the most general level of universality, in which we attempt to create a single *input-agnostic* perturbation which is able to fool any possible input. This particular level is the only one investigated in all previous works on audio universal adversarial perturbations. In this case, the training set is composed of 100 samples per class, forming a total of 1200 training samples.

Results for Level 3 perturbations are shown in Table IV. Comparing the FR achieved in each class, it can be seen that the perturbation effectively fools a considerable percentage of samples in the majority of the classes. In particular, in 2 out of 12 classes the FR is at least 70% in the training set, and for 8 out of 12 classes at least 60%. In the validation set, the FR is at least 60% for 6 classes. As for previous levels, the mean distortion is below -32dB in all the cases except for the class *silence*. However, the total FR is reduced because of the low results obtained in classes *silence* and *left*, achieving a total final FR near 50% in both training and validation sets. Although room for improvement exists regarding the effectiveness of the perturbations, it is noteworthy that the attacks generated imply a noticeable drop of the accuracy of the model, of approximately 50% in both sets. In comparison to recent works in which universal perturbations are generated for the speech command classification task [7], [8], although the generated perturbations achieved a lower fooling ratio, the distortion levels are also considerably lower.

TABLE III
RESULTS OBTAINED IN LEVEL 2 UNIVERSAL PERTURBATIONS

Pair	FR% - Target Model A						FR% - Target Model B						Mean dB
	Train			Valid			Train			Valid			
	Total	C1	C2	Total	C1	C2	Total	C1	C2	Total	C1	C2	
C1: no C2: go	71.20	68.80	73.60	68.35	72.52	64.31	41.36	37.50	45.52	49.04	47.90	50.34	-46.37
C1: up C2: off	67.50	70.40	64.00	62.03	62.40	61.58	28.14	31.70	24.69	31.94	34.10	29.39	-45.31
C1: yes C2: left	36.90	73.80	0.00	36.78	71.32	0.00	29.12	58.79	0.00	32.20	64.60	0.00	-38.83
C1: unk. C2: stop	48.90	55.20	42.60	47.41	50.55	45.19	15.21	10.57	19.20	21.54	16.90	25.13	-41.16
C1: silence C2: unk.	43.30	19.40	67.20	35.60	16.75	63.84	27.56	24.65	29.63	32.25	21.65	40.69	-36.32
C1: no C2: up	70.90	71.60	70.20	65.95	70.86	62.15	7.86	3.97	11.79	9.12	6.89	11.03	-42.04

TABLE IV
RESULTS OBTAINED IN LEVEL 3 UNIVERSAL PERTURBATIONS

Class	FR% Target Model A		FR% Target Model B		Mean dB
	Train	Valid	Train	Valid	
Silence	26.00	18.72	36.07	35.50	-23.88
Unknown	62.00	61.62	44.30	49.31	-36.66
Yes	62.00	66.24	63.33	64.14	-37.56
No	61.00	66.23	44.21	51.20	-36.74
Up	70.00	61.89	52.08	50.26	-38.18
Down	52.00	44.82	22.58	15.53	-37.97
Left	0.00	0.00	0.00	0.26	-37.60
Right	60.00	55.84	35.05	38.99	-37.58
On	60.00	57.39	26.88	28.24	-36.42
Off	73.00	68.60	43.16	48.48	-38.27
Stop	45.00	43.38	29.17	33.42	-38.25
Go	60.00	63.34	44.57	51.72	-38.01
Total	52.58	46.64	36.53	38.70	-36.43

D. Transferability

The transferability levels of the generated universal perturbations is described in Tables II, III and IV, for Level 1, Level 2 and Level 3 universal perturbations respectively. Similarly to the original target model, the effectiveness of the perturbations is considerably higher for some classes or subsets of classes than for others. In Level 1 perturbations, for 4 out of 12 classes the FR is above 30% in the validation set. In Level 2 perturbations, for 4 out of 6 pairs of classes the FR is above 30%. Finally, for Level 3 perturbations, although the total FR in the validation set is 38.70%, for 4 out of 12 classes the FR is above 50%, and for 9 out of 12 classes above 30%. As can be seen, the generalization of the perturbations across models is generally higher as the universality level increases, which may reflect that, when a large number of classes are considered, the perturbations are able to generalize better not only across classes, but also across target models.

E. Discussion

The analysis carried out for different levels of universality revealed some interesting insights about universal perturbations in this domain: 1) it is considerably easier to create perturbations for some subsets of classes than for others, 2) while generating Level 1 universal perturbations is considerably more feasible than Level 2 or Level 3 perturbations, generating Level 3 perturbations is not necessarily more difficult than Level 2 perturbations and 3) the transferability of the attacks increases as the universality level increases. All these conclusions could be used as a basis to better study the robustness of the model against adversarial attacks or to provide a theoretical explanation of universal perturbations.

VII. DEEPER EVALUATION OF THE AMOUNT OF DISTORTION

To measure the loudness of an audio signal x of length d, previous related works [6]–[8], [17] make use of the maximum decibel level of the signal ($dB_{x,max}$ metric). We have shown in Section VI that, under this criterion, our perturbations were far below the threshold of -32dB, and therefore, assumed to be quasi-imperceptible according to the standards established by convention. The drawback of considering only the maximum decibel level of the signals to compute the distortion is that this metric can be highly sensitive to peaks in the intensity of the audio, which may lead to results that do not adequately represent the overall loudness level of the audio. To obtain a more accurate and robust evaluation of adversarial examples, we propose using, in addition to this distortion metric, another one based on the mean loudness of the signal $dB_{mean}(x) = 20 \cdot log_{10}\left(\frac{1}{d}\sum_{i=1}^{d}|x_i|\right)$. As in the previous case, the distortion level of a signal v with respect to another reference signal x is measured as $dB_{x,mean}(v) = dB_{mean}(v) - dB_{mean}(x)$.

Besides, previous works compute the distortion by applying the metrics to the entire signals. To provide a more detailed evaluation of the distortion, we propose analyzing the perturbation for different parts of the signals independently, partitioned according to the information they convey. In particular, we will consider two different parts in each audio signal: the *vocal* part and the *background* part. As both parts are significantly different, and because the distortion will be measured based on aggregated features of the audio signals, evaluating these metrics in the entire signal could lead to misleading results. Thus, we will study the distortion in these two parts independently, to obtain more informative results. To the best of our knowledge, a similar type of analysis has not been previously reported in the related literature.

Being $x = \{x_1, \ldots, x_d\}$ an audio signal, the vocal part will be delimited by the continuous range $\{x_a, \ldots, x_b\}$ containing 95% of the energy of the signal, that is, $\frac{\sum_{j=a}^{b} x_j^2}{\sum_{i=1}^{d} x_i^2} \approx 0.95$.

Thus, we will assume that the intervals $\{x_1, \ldots, x_{a-1}\}$ and $\{x_{b+1}, \ldots, x_d\}$ will be composed just of background noise. The amount of distortion computed using this approach is presented in Fig. 1, represented for the Level 3 universal perturbations. According to the results, the amount of distortion in the vocal part is acceptable for the great majority of the samples for the two metrics considered. However, the distortion in the background part is above the threshold of -32dB for a considerably high proportion of samples according to the metric $dB_{x,mean}(v)$, which indicates that the perturbation may be highly detectable outside the vocal part. We encourage the reader to listen to some adversarial examples, to empirically verify that the perturbations are actually detectable.[8] These results strengthen our proposal about the need to employ more rigorous approaches in order to measure and threshold the distortion produced by the adversarial perturbations in a more representative way.

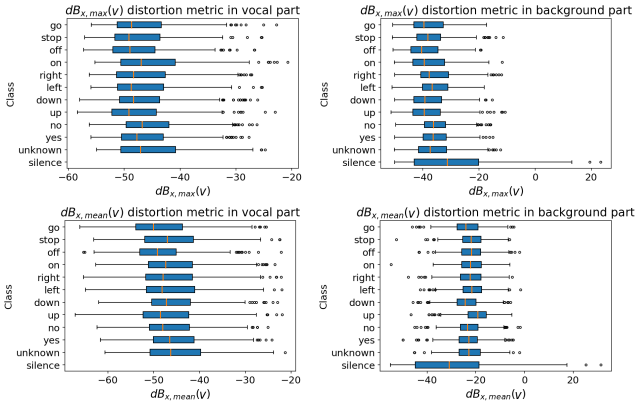


Fig. 1. Distortion level of the generated Level 3 universal perturbations, evaluated in the validation set. For each audio, the distortion has been measured in the background part as well as in the vocal part, except for the class silence, as it is only composed of background noise. Results are averaged for the 5 experiments carried out.

VIII. Conclusions

In this work the feasibility of generating universal adversarial perturbations in the task of speech command classification has been addressed. To deal with this question, we have introduced the concept of N-class universality as a theoretical framework for the study of universal perturbations, which is based on the number of target classes that the universal attack is designed to inhibit. This framework has been used to determine the feasibility of generating universal perturbations according to different levels of universality. The reported results demonstrate that effective universal adversarial attacks can be generated for speech command classification models, although improvement is needed in order to consider the perturbations fully universal. The results also reflected that, although the effectiveness of the perturbations was reduced in new target models, they were transferable across models to a significant extent.

Finally, we have shown that, although the distortion levels of the perturbations are acceptable according to the standard evaluation approach employed by convention in previous works, under a more rigorous evaluation in which different parts of the signals are inspected independently, the perturbations were detectable outside the vocal parts corresponding to the speech command. This fact should be addressed in future research to achieve less detectable perturbations.

References

[1] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," in *2nd International Conference on Learning Representations*, 2014. [Online]. Available: http://arxiv.org/abs/1312.6199

[2] Y. Gal and L. Smith, "Idealised bayesian neural networks cannot have adversarial examples: Theoretical and empirical study," *arXiv preprint arXiv:1806.00667*, 2018.

[3] N. Akhtar and A. Mian, "Threat of adversarial attacks on deep learning in computer vision: A survey," *IEEE Access*, vol. 6, pp. 14 410–14 430, 2018.

[4] S.-M. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, and P. Frossard, "Universal adversarial perturbations," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 86–94.

[8] https://vadel.github.io/audio_adversarial/UniversalPerturbations.html

[5] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "DeepFool: a simple and accurate method to fool deep neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2574–2582.

[6] P. Neekhara, S. Hussain, P. Pandey, S. Dubnov, J. McAuley, and F. Koushanfar, "Universal adversarial perturbations for speech recognition systems," in *Interspeech 2019*, 2019, pp. 481–485.

[7] S. Abdoli, L. G. Hafemann, J. Rony, I. B. Ayed, P. Cardinal, and A. L. Koerich, "Universal adversarial audio perturbations," *arXiv preprint arXiv:1908.03173*, 2019.

[8] Y. Xie, Z. Li, C. Shi, J. Liu, Y. Chen, and B. Yuan, "Enabling fast and universal audio adversarial attack using generative model," *arXiv preprint arXiv:2004.12261*, 2020.

[9] Z. Li, Y. Wu, J. Liu, Y. Chen, and B. Yuan, "AdvPulse: Universal, synchronization-free, and targeted audio adversarial attacks via subsecond perturbations," in *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*, 2020, pp. 1121–1134.

[10] Y. Xie, C. Shi, Z. Li, J. Liu, Y. Chen, and B. Yuan, "Real-time, universal, and robust adversarial attacks against speaker recognition systems," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2020, pp. 1738–1742.

[11] J. Rony, L. G. Hafemann, L. S. Oliveira, I. B. Ayed, R. Sabourin, and E. Granger, "Decoupling direction and norm for efficient gradient-based L2 adversarial attacks and defenses," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4317–4325.

[12] P. Warden, "Speech commands: A dataset for limited-vocabulary speech recognition," *arXiv preprint arXiv:1804.03209*, 2018.

[13] M. Alzantot, B. Balaji, and M. Srivastava, "Did you hear that? Adversarial examples against automatic speech recognition," *arXiv preprint arXiv:1801.00554*, 2018.

[14] T. Du, S. Ji, J. Li, Q. Gu, T. Wang, and R. Beyah, "SirenAttack: Generating adversarial audio for end-to-end acoustic systems," in *Proceedings of the 15th ACM Asia Conference on Computer and Communications Security*, 2020, pp. 357–369.

[15] Z. Yang, B. Li, P.-Y. Chen, and D. Song, "Characterizing audio adversarial examples using temporal dependency," *arXiv preprint arXiv:1809.10875*, 2018.

[16] T. N. Sainath and C. Parada, "Convolutional neural networks for small-footprint keyword spotting," in *Interspeech 2015*, 2015, pp. 1478–1482.

[17] N. Carlini and D. Wagner, "Audio adversarial examples: Targeted attacks on speech-to-text," in *2018 IEEE Security and Privacy Workshops*, 2018, pp. 1–7.

[18] "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: http://tensorflow.org/

[19] S. W. Smith *et al.*, "The Scientist and Engineer's Guide to Digital Signal Processing," 1997.

Automatic Labeling Using QA

Mª Eugenia Taillefer
Aifunded
Madrid, Spain
eugenia.taillefer@aifunded.es

Manuel Baena-García, Rafael Morales-Bueno
Escuela Técnica Superior de Ingeniería Informática
Universidad de Málaga
Málaga, Spain
mbaena@uma.es, rmorales@uma.es

***Abstract*—Currently Question Answering (QA) systems have evolved enormously. QA systems are computationally expensive to train. Natural Language Processing (NLP) has become the AI area with the greatest potential, especially in multi-language. On the other hand, transformers have brought AI in general and NLP in particular closer to the market, closing the gap between research, development and innovation.**

To effectively apply AI techniques, data are needed and they are not always available. One of the most time-consuming tasks is data Labeling. Industry needs efficient solutions to apply AI through the whole chain, because it always needs to feedback the system and control concept drift.

In this paper we propose to use the current state of the art in NLP, transformers and question answering to facilitate the extraction of data from text-based applications in Spanish, in particular in the legal sector. This allows reducing the costs of both digitization and application of AI in the market and in specific information domains.

Results show that we achieve 77% Exact Match score without any specific domain treatment, and 90% Exact Match score using specialized domain filters.

Keywords— Question answering systems, QA, Natural language processing, NLP, Labeling, Spanish.

I. INTRODUCTION

Question Answering (QA) systems are powerful platforms for automatically answering questions asked by humans in natural language using a collection of natural language documents. With the efforts from academic research, Question Answering is a growing research field worldwide. The demand for this kind of system increases day by day since it delivers short, precise and question-specific answers. [2]

In machine learning, data labeling is the process of identifying raw data (images, text files, videos, etc.) and adding one or more meaningful and informative labels to provide context so that a machine learning model can learn from it. Data Labeling is required for natural language processing. [3]

These short, precise and question-specific answers from QA fit perfectly to label data.

In summary, the contributions we make are the following:

— We apply an existing, emerging and mature technology such as open domain QA based on publicly available data to legal sector domain with private data. There is no Spanish dataset for legal domain.

— We use this method to fill relevant data into databases in order to apply other machine/deep learning processes.

— We demonstrate that this method is cost-efficient and good enough for industry purposes.

— We combine expertise from legal sector and Deep Learning domain to find the best labeling tool, increasing quality of results and minimizing human costs.

II. OBJECTIVE

This study aims at applying open-domain pretrained QA tools to labeling data in Spanish language and to the specific-domain of legal sector.

III. BACKGROUND

In this section we briefly describe the tools used to carry out the experiment. There are several components in this proposal.

A. Components

— QA systems: English or multi-language; and open or close domain

— QA systems in Spanish

— Labeling systems

— Transformers

1) QA systems

Question Answering systems enable users to retrieve exact answers for questions posed in natural language. [2] Question Answering systems can be categorized into open or close domain, and questions can be divided into factoid or non-factoid.[7] QA open domain (ODQA) imitate human intelligence, questions and answers come from any domain. Systems are trained from large datasets like Wikipedia as source knowledge.

QA closed domain (CDQA) exploit domain-specific knowledge using a trained domain-specific dataset. Questions belong to this specific-domain, as well as answers.

Factoid questions are defined as questions that have a short answer (typically a noun phrase or a simple verb phrase) or an enumeration of such short answers.[8] Factoid Questions are

like W5H1 questions fired possibly at a reference text, or questions that answer to who, what, where, when, why, how, like real Wikipedia facts. [9] Answers can be named entities or numbers.[6]

Non-factoid questions are defined as open-ended questions that require complex answers, like descriptions, opinions, or explanations, which are mostly passage-level texts.[6]

Currently, we can finetune pretrained open-domain QA systems in specific domain, to reduce human and computational costs.

2) QA systems in Spanish

Recently a Spanish open-domain QA Dataset, SQuAD-es v2.0, has been provided [1]. It consists of translating SQuAD dataset versions 1 and 2, and providing a method to answer in Spanish. It uses a translate-align-and retrieve method.

The Stanford Question Answering Dataset (SQuAD) is a reading comprehension dataset, consisting of questions posed by crowd workers on a set of Wikipedia articles. The answer to every question is a segment of text (a span) from the corresponding reading passage. SQuAD 2.0 includes unanswerable questions, its public leader-board is on the SQuAD website.[10]

3) Labeling systems

In-house labeling methods are human time consuming and have a reliable quality. Automatic labeling is computationally expensive and has a poor quality. [11]

4) Transformers

Transformers are encoder-decoder neural networks with attention. They excel in translation tasks. And they are capable of abstracting the contained information, more effectively than recurrent neural networks. And they allow to transfer knowledge, so large trained NN can be transferred to other systems.

IV. Method

Working for the legal sector we identified the need of Spanish legal datasets, powerful and cost-effective tools to label data and question-answering systems to retrieve information in natural language. This information is already stored in available files, but staff usually ask for them outside the company, delaying the process.

A. Hypothesis

Possible results of this experiment of labeling data via QA are the following:

1. not valid: results are poor in quality and don't match the correct answer.
2. semiautomated: results are good but they are not perfect. We preprocess the pairs question-answers in batch, and use human interaction to correct the results and save them in the database. We provide a web page with a list of results: score, question, answer and highlight the answer in the document containing the answer.
3. automatic: results are correct. Just applying some question-specific filters to ignore incorrect answers, we can record the results in the database.

In this paper we use SQuAD-es v2.0 dataset. [1] We have a tuple of three parts: passage, question, and answer. The QA system adds the initial and end position of the answer in the passage and a score.

B. Tasks

Task 1.- Converting documents to text from pdf, tiff, doc, eml, jpg, txt, rtf, etc. Images are converted to text in Spanish with OCR tools like Tesseract.

In this experiment a total amount of 1280 TIF files have been processed and anonymized. These selected files are the worst quality files in the legal cases. Adding pdf and digital documents will improve the quality of the system. Better quality in documents improves accuracy.

Task 2.- Files distribution. Files for a legal case (case) are stored in a unique folder. Finding the answer to a question in a case, requires feeding the QA system with all text files in the folder. So, we can sort court briefs by date and case.

Task 3.- Question-Answering. Posing a question in Spanish to the QA system to retrieve relevant data from a case. In this experiment the question is "How much are interests and procedural costs?". We get several answers per case. We set a score threshold and ignore answers below it.

Task 4.- Evaluation Method. To evaluate our models, we use the standard SQuAD performance metrics: Exact Match (EM) score and F1 score.

Exact Match: A binary measure of whether the system output matches the ground truth answer exactly, character by character.

F1: Harmonic mean of precision and recall [5].

$$\text{F1 score} = (2\times\text{prediction}\times\text{recall}) / (\text{precision} + \text{recall}) \quad (1)$$

$$\text{precision} = \text{num\_shared\_words}/\text{num\_words\_prediction} \quad (2)$$

$$\text{recall} = \text{num\_shared\_words}/\text{num\_words\_ground\_truth} \quad (3)$$

$$\text{num\_shared\_words} = \text{set\_intersection}(\text{set\_words}(\text{prediction}), \text{set\_words}(\text{ground\_truth})) \quad (4)[4]$$

As the answer is just a word (a quantity without currency), EM and F1 score are the same metric. So, we will use EM score from now on.

We group the answers by reported amount, set vote as the number of equal answers, and set score to the maximal score in group. We iterate over all the answers to a question, and select the answer with majority of votes. Score is used as second selection criteria in the event of a tied vote. See Table I and II. Let's see an example.

TABLE I. EXAMPLE OF ANSWER TO A QUESTION IN A CASE.

Question:				How much is interests and procedural costs?
Case file:				0178002
Answers	**score**	**file**	**correct**	**passage**
16026.30	0.59	abc.txt	Yes	It remains to pay the amount of **16026.30** eur in concept of interests and procedural costs
16026.30	0.57	bcd.txt	Yes	by the amount of **16026.30** eur for interests and procedural costs we inform the court
86.78	0.21	cde.txt	No	Attached I send you a return order from BANK in favor of our client, which for the amount of **86.78** eur I have withdrawn from the Court as a INTERESTS AND PROCEDURAL COSTS.

TABLE II. ANSWER GROUPS

answers	**votes**	**score**
16026.30	2	0.59
86.78	1	0.21

I. RESULTS

In a set of 30 cases (1280 files) a 67% of exact match with open-domain Spanish QA system is achieved.

Applying majority vote algortihm increases EM to 77%. This is due probably to the fact that relevant data is referenced in more than one document. If there is any conversion error in one of them, it will be solved by majority.

We analyze the encountered errors. See Table III. Total and quality errors mainly come from poor quality characters in original documents, that usually remove words when converted to text, loosing passage coherence. Applying a filter to correct "total" errors we achieve a 90% EM score.

Note that training this system to not answer if the information is not contained in the passage could solve total filters plus zero answer. This is a correction of 5 answers, giving an EM score of 93%. See Table IV. The remaining 2 answers are evident for humans as incorrect, last two in Table III.

TABLE III. ERROR ANALYSIS AND REASON

total 7	**error type**	**Adding filter to ignore answers if:**
4	total	ignore answers with total debt: principal + interests + procedural costs
1	quality	check if answer is zero: info contained but rarely expressed and poor-quality file
1	quality	check if answer contains just decimal numbers: Justice Ministry stamp is overlapping the amount
1	quality	dot decimal is missing: after converting to text, poor quality original file

TABLE IV. RESULTS WITH OPEN-DOMAIN QA AND APPLYING QUESTION SPECIFIC FILTERS

Total cases 30 (1280 files)	**correct**	**incorrect**	**EM**
best answer (sorted by score)	20	10	67%
majority vote (sorted by max score in vote group)	23	7	77%
majority vote plus "total" filter	27	3	90%

I. CONCLUSION AND FUTURE WORK

Application of QA to new areas improve results, reduces costs and allows companies to digitalize and apply AI to a natural language huge information volume in Spanish. Scanned documents achieve excellent results. QA is a good proposal cause also helps non-technical staff to retrieve stored information. Progress in QA and Spanish is a rapid evolving area. We achieved 90% in EM score in labeling specific data, provided a method to label, and to store them. We add our solution to empower in house trained staff and Deep Learning solutions to automatic labeling process. It brings quality reducing time consuming from humans. QA labeling data is automatable.

In order to achieve better results, we propose to train the network with domain specific passage, recurrent questions and answer tuples, providing 5 truth answers, and enabling null answers. Search engines to locate the most relevant pre-classified documents can improve training and inference time. Long QA systems provide large and short answers. They could be a good solution to build a training dataset in specific domain.

REFERENCES

[1] Carrino, Casimiro & Costa-jussà, Marta & Fonollosa, José. (2019). "Automatic Spanish Translation of the SQuAD Dataset for Multilingual Question Answering," in International Conference on Language Resources and Evaluation. "LREC 2020: 12th International Conference on Language Resources and Evaluation: Marseílle, France: May 13-15, 2020: conference proceedings". Paris: European Language Resources

Association (ELRA), 2020, p. 5515-5523. ISBN 979-10-95546-34-4, http://hdl.handle.net/2117/329270

[2] Calijorne Soares, Silva Parreiras. "A literature review on question answering techniques, paradigms and systems," in Journal of King Saud University - Computer and Information Sciences, Volume 32, Issue 6, pp 635-646, 2020, doi: 10.1016/j.jksuci.2018.08.005, https://www.sciencedirect.com/science/article/pii/S131915781830082X (2020)

[3] "What is data labeling for machine learning? ", AWS, https://aws.amazon.com/es/sagemaker/groundtruth/what-is-data-labeling/ (accessed May 6, 2021).

[4] Melanie Beck, "Evaluating BERT on SQuAD", Cloudera Fast Forward, https://colab.research.google.com/github/fastforwardlabs/ff14_blog/blob/master/_notebooks/2020-06-09-Evaluating_BERT_on_SQuAD.ipynb, (accessed May 6, 2021).

[5] Schwager, Solitario. "Question and Answering on SQuAD 2 . 0 : BERT Is All You Need.", 2019, https://web.stanford.edu/class/archive/cs/cs224n/cs224n.1194/reports/default/15812785.pdf.

[6] Hashemi H., Aliannejadi M., Zamani H., Croft W.B. (2020) ANTIQUE: A Non-factoid Question Answering Benchmark. In: Jose J. et al. (eds) Advances in Information Retrieval. ECIR 2020. Lecture Notes in Computer Science, vol 12036. Springer, Cham. doi: 10.1007/978-3-030-45442-5_21

[7] Sabharwal, Agrawal. *Hands-on Question Answering Systems with BERT*. Apress. India, 2021, doi: 10.1007/978-1-4842-6664-9

[8] Agichtein, Cucerzan, Eric Brill. (2005). Analysis of factoid questions for effective relation extraction. In Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR '05). Association for Computing Machinery, USA, pp 567–568.

[9] Rob van Zoest, "Natural Language Processing: What is Non-factoid question answering?", Quora.com, https://www.quora.com/Natural-Language-Processing-What-is-Non-factoid-question-answering, (accessed May 6, 2021).

[10] Sebastian Ruder, "NLP progress", http://nlpprogress.com/english/question_answering.html, (accessed May 6, 2021).

[11] Hengtee Lim. "5 Approaches to Data Labeling for Machine Learning Projects". Lionbridge.ai. https://lionbridge.ai/articles/5-approaches-to-data-labeling-for-machine-learning-projects/ (accessed May 6, 2021).

Reemplazo de la función de pooling de Redes Neuronales Convolucionales por combinaciones lineales de funciones crecientes

1º Iosu Rodríguez-Martínez
dept. de Estadística, Informática y Matemáticas
Universidad Pública de Navarra
Pamplona, España
iosu.rodriguez@unavarra.es

2º Julio Lafuente
dept. de Estadística, Informática y Matemáticas
Universidad Pública de Navarra
Pamplona, España
lafuente@unavarra.es

3º Mikel Sesma
dept. de Estadística, Informática y Matemáticas
Universidad Pública de Navarra
Pamplona, España
mikel.sesma@unavarra.es

4º Francisco Herrera
dept. de Ciencias de la Computación e Inteligencia Artificial
Universidad de Granada
Granada, España
herrera@decsai.ugr.es

5º Pablo Ursúa-Medrano
dept. de Estadística, Informática y Matemáticas
Universidad Pública de Navarra
Pamplona, España
pablo.ursua@unavarra.es

6º Humberto Bustince
dept. de Estadística, Informática y Matemáticas
Universidad Pública de Navarra
Pamplona, España
bustince@unavarra.es

***Abstract*—Las redes convolucionales llevan a cabo un proceso automático de extracción y fusión de características mediante el cual obtienen la información más relevante de una imagen dada. El proceso de submuestreo mediante el cual se fusionan características localmente próximas, conocido como "pooling", se lleva a cabo tradicionalmente con funciones sencillas como el máximo o la media aritmética, ignorando otras opciones muy populares en el campo de la teoría de agregaciones. En este trabajo proponemos reemplazar dichas funciones por otra serie de órdenes estadísticos, así como por la integral de Sugeno y una nueva generalización de la misma. Además, basándonos en trabajos que emplean la combinación convexa del máximo y la media, presentamos una nueva capa que permite combinar varias de las nuevas agregaciones, mejorando sus resultados individuales.**

***Index Terms*—Redes Neuronales Convolucionales, Función de pooling, Combinaciones Lineales, Integral de Sugeno Generalizada**

I. Introducción

Las redes neuronales convolucionales (CNN, por sus siglas en inglés) han cobrado gran popularidad desde que Krizhevsky et al. ganaran la competición ImageNet de 2012 mediante su modelo AlexNet [1], y actualmente constituyen el estado del arte en aplicaciones de tratamiento de imagen.

Inicialmente, el foco se puso en el desarrollo de arquitecturas más y más profundas que buscaban mejorar su capacidad de clasificación mediante la adición de más parámetros [2], [3]. Actualmente, el objetivo contrario ha crecido también en interés, puesto que el coste de entrenar estos modelos puede resultar prohibitivo en ciertos dominios de aplicación (p. e. "smartphones" o vehículos autónomos). Con este objetivo en mente, se han ido introduciendo arquitecturas más modestas aunque competitivas [4]–[6], y estrategias como la transferencia de conocimiento o el podado de modelos se han postulado como estrategias valiosas [7], [8].

Pese a la cantidad de investigación desarrollada en torno a este campo apodado "Deep Learning", las CNNs siguen operando en base a las funciones clásicas en torno a las cuales fueron diseñadas: la convolución, que extrae características locales a partir de una imagen; y el "pooling", que agrega características próximas espacialmente para reducir la dimensionalidad de la salida generada. A pesar de que, con respecto a esta última, se han presentado algunas alternativas [9]–[11], la mayoría de los modelos siguen empleando el máximo o la media aritmética.

En [12] los autores presentan la combinación convexa del máximo y la media aritmética como una función de pooling que aprovecha la información proporcionada por cada una mejorando su comportamiento individual. Creemos que este hecho, combinado a los avances en teoría de agregación de los últimos años, pueden ayudarnos a generar funciones que superen el rendimiento de las opciones tradicionales.

En este artículo, reemplazamos la función de pooling de

varios modelos de CNN por combinaciones lineales crecientes de estas funciones así como de otros órdenes estadísticos y de la integral de Sugeno, que permite modelar la interacción entre los valores por medio del concepto de medida difusa. Nuestros resultados demuestran que combinar distintas funciones puede mejorar su comportamiento individual y que, en particular, una generalización de esta integral tiende a obtener los mejores resultados cuando se combina con la media aritmética o el máximo.

El artículo se estructura de la siguiente forma: en la sección 2 se recuerdan algunos conceptos relacionados con las CNNs y las funciones de agregación propuestas; la sección 3 compone el estudio teórico que demuestra las condiciones a cumplir por nuestras combinaciones con el fin de comportarse como una función creciente; en la sección 4 demostramos la utilidad del método al emplearlo en varias arquitecturas de Deep Learning estándares en la literatura; la sección 5 cierra el artículo con algunas conclusiones y líneas futuras a explorar.

II. Preliminares

A. Redes Neuronales Convolucionales

Las CNNs son un tipo de red neuronal artificial diseñada para tratar con datos cuya información local es relevante, como series temporales [13] o imágenes [1]. Estos modelos se componen de dos partes diferenciadas: una primera que se encarga de extraer las características más relevantes a partir de los datos de entrada de manera automatizada y una segunda, que emplea estas características para resolver un problema específico de su dominio de aplicación. En el caso a tratar, el de la clasificación de imagen, se tratará de un clasificador que, para una imagen de entrada, devolverá la probabilidad de pertenencia a varias posibles clases predefinidas.

Es el primer proceso el que da su nombre a las CNN, dado que la extracción de características se produce mediante la convolución de varios filtros de características sobre una entrada dada. Por entrada entendemos una matriz $X \in \mathbb{R}^{h\times w\times c}$, que puede representar, bien una imagen de tamaño $h \times w$, donde cada pixel $X_{ij} \in \mathbb{R}^c$ cuenta con c canales de color ($c = 1$ para imágenes en gris, $c = 3$ para imágenes en color), o bien una imagen de características, donde cada canal $X^d \in \mathbb{R}^{h\times w}$ representa la presencia o ausencia de una característica concreta en las distintas partes de la imagen. Cada uno de esos canales habrá sido producido, a su vez, por la convolución de otro filtro de características que representará una característica concreta sobre otra entrada anterior.

Para ser más precisos, la salida producida por un filtro de características $W \in \mathbb{R}^{k_1\times k_2\times c}$ para una ventana $k_1 \times k_2$ (con $k_1, k_2 \in \mathbb{N}$ números impares) de una entrada $X \in \mathbb{R}^{h\times w\times c}$ centrada en el pixel $X_{m,n}$ se calcula mediante la siguiente expresión:

$$Y_{m,n} = \sum_{d=1}^{c}\sum_{j=1}^{k_2}\sum_{i=1}^{k_1} W_{ij}^d \cdot X^d_{m-1+i-\frac{k_1-1}{2},n-1+j-\frac{k_2-1}{2}} \quad (1)$$

Los valores de estos filtros se "aprenden" de manera análoga a los pesos de un perceptrón multicapa: se inicializan aleatoriamente y sus valores se optimizan de manera iterativa mediante alguna variación del algoritmo de descenso por gradiente, comúnmente el descenso por gradiente estocástico [14].

Las CNN se componen de varias capas convolucionales (además de otros tipos de capas) que operan secuencialmente sobre los datos de entrada tal que la salida generada por la capa l-ésima se introduce como entrada a la capa $(l+1)$-ésima. Cada capa convolucional está formada por c_l filtros de características que se aplican a la entrada recibida, generando c_l imágenes de características que se envían a la siguiente capa.

Pese a que el objetivo del extractor de características es el de reducir la imagen, el proceso anterior tiene el efecto contrario. Para lograr esta reducción se añaden capas de pooling en distintos puntos de la red. Cuando una capa de pooling recibe una entrada $X \in \mathbb{R}^{h\times w\times c}$, cada canal independiente se separa en ventanas disjuntas de tamaño $k_1 \times k_2$. Los valores de cada ventana se agregan mediante alguna función, tradicionalmente el máximo o la media aritmética.

B. Funciones crecientes

Aquí recordamos y fijamos la notación para algunas funciones crecientes que usaremos más adelante.

De ahora en adelante, asumimos que $2 \leq n \in \mathbb{N}$, $1 \leq r \in \mathbb{N}$.

Decimos que una función $\mathbf{A}\colon \mathbb{R}^n \to \mathbb{R}$ es creciente si $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$, $\mathbf{x} \leq \mathbf{y}$ implica que $\mathbf{A}(\mathbf{x}) \leq \mathbf{A}(\mathbf{y})$

1) Estadísticos de orden:

Notación II.1. Fijamos $N = \{1, \ldots, n\}$ y Σ_n como el grupo de todas las posibles permutaciones de N. Si $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{R}^n$ y $\sigma \in \Sigma_n$, lo denotamos mediante $\mathbf{x}_\sigma = (x_{\sigma(1)}, \ldots, x_{\sigma(n)})$; si $x_{\sigma(1)} \leq \cdots \leq x_{\sigma(n)}$, lo denotamos mediante $\sigma \in \mathbf{x}_{(\nearrow)}$.

Sea $r \in \{1, \ldots, n\}$. Denotamos mediante $\mathbf{OS}_r$ al r-ésimo estadístico de orden, esto es, la función $\mathbf{OS}_r\colon \mathbb{R}^n \to \mathbb{R}$ dada por, si $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{R}^n$, $\mathbf{OS}_r(\mathbf{x}) = x_{\sigma(r)}$, donde $\sigma \in \mathbf{x}_{(\nearrow)}$. En particular $\mathbf{OS}_n$ es el máximo y $\mathbf{OS}_1$ el mínimo.

2) Media aritmética: Denotamos mediante $\mathbf{AM}$ a la media aritmética, esto es la función $\mathbf{AM} : \mathbb{R}^n \to \mathbb{R}$, dada por $\mathbf{AM}(\mathbf{x}) = (x_1 + \cdots + x_n)/n$ si $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{R}^n$.

3) Integral de Sugeno y generalizaciones:

Definición II.1. Una medida difusa en N es una relación $\nu\colon 2^N \to [0, +\infty)$ tal que

1) $\nu(\emptyset) = 0$ y
2) $S \subseteq T \subseteq N$ implica $\nu(S) \leq \nu(T)$.

Definición II.2. La *integral de Sugeno* asociada a una medida difusa ν es la función

$$\mathbf{S}_\nu\colon \mathbb{R}^n \to \mathbb{R}$$

dada por, para $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{R}^n$,

$$\mathbf{S}_\nu(\mathbf{x}) = \mathbf{max}\left(\mathbf{min}(x_{\sigma(1)}, \nu(N_1^\sigma)), \ldots, \mathbf{min}(x_{\sigma(n)}, \nu(N_n^\sigma))\right),$$

donde $\sigma \in \mathbf{x}_{(\nearrow)}$ y $N_i^\sigma = \{\sigma(i), \ldots, \sigma(n)\}$.

Una posible extensión que utilizaremos más adelante fue presentada en [15].

III. Combinación de funciones crecientes

A. Construcción de funciones crecientes mediante combinaciones lineales de funciones crecientes

A continuación estudiamos combinaciones lineales de funciones crecientes de $\mathbb{R}^n \to \mathbb{R}$, en algunos casos de $[0,+\infty)^n \to [0,+\infty)$.

Fijamos $\mathbf{0} = (0,\dots,0)$, $\mathbf{1} = (1,\dots,1)$, $\mathbf{e}_i = (0,\dots,\underset{i}{1},\dots,0)$.

Notación III.1. Sean $\mathbf{A}_1,\dots,\mathbf{A}_r\colon \mathbb{R}^n \to \mathbb{R}$ funciones crecientes. Fijamos

$$\mathcal{I}(\mathbf{A}_1,\dots,\mathbf{A}_r) = \\ = \{(\alpha_1,\dots,\alpha_r) \in \mathbb{R}^n \mid \sum_{i=1}^{r} \alpha_i \mathbf{A}_i\colon \mathbb{R}^n \to \mathbb{R} \text{ es una función creciente}\}.$$

B. Combinaciones de estadísticos de orden y la media aritmética

Para que la combinación lineal de estadísticos de orden sea una función creciente, se debe cumplir la siguiente propiedad

Proposición III.2. *Sea $i_1,\dots,i_r \in N$, $i_1 < \cdots < i_r$. Entonces*

$$\mathcal{I}(\mathbf{OS}_{i_1},\dots,\mathbf{OS}_{i_r}) = \{(\alpha_1,\dots,\alpha_r) \mid \alpha_1,\dots,\alpha_r \geq 0\}$$

Por definición $\mathbf{AM} = \frac{1}{n}(\mathbf{OS}_1 + \cdots + \mathbf{OS}_n)$, tenemos

Proposición III.3. *Sea $i_1,\dots,i_r \in N$, $i_1 < \cdots < i_r$, $r < n$. Entonces*

$$\mathcal{I}(\mathbf{AM},\mathbf{OS}_{i_1},\dots,\mathbf{OS}_{i_r}) = \\ = \{(\alpha,\beta_1,\dots,\beta_r) \mid \alpha,\alpha+n\beta_1,\dots,\alpha+n\beta_r \geq 0\}.$$

Análogamente

Proposición III.4. *Tenemos*

$$\mathcal{I}(\mathbf{AM},\mathbf{OS}_1,\dots,\mathbf{OS}_n) = \\ = \{(\alpha,\beta_1,\dots,\beta_n) \mid \alpha+n\beta_1,\dots,\alpha+n\beta_n \geq 0\}.$$

C. Combinaciones con la Integral de Sugeno

Escribimos $\nu_S = \nu(S)$ para $S \subseteq N$, $\mathbf{x}_\sigma = (x_{\sigma(1)},\dots,x_{\sigma(n)})$, $\nu_i^\sigma = \nu(N_i^\sigma)$, $N_i = \{i,\dots,n\}$ y $\nu_i = \nu(N_i)$. Por lo tanto $\mathbf{S}_\nu(\mathbf{x}) = \bigvee_{i=1}^n (x_{\sigma(i)} \wedge \nu_i^\sigma)$. Sabemos que

$$S_\nu(x_1,\dots,x_n) = \bigvee_{S\subseteq N} [\nu_S \wedge (\bigwedge_{i\in S} x_i)].$$

Lemma III.5. *Si $S \subseteq N$, entonces existe $\sigma \in \Sigma_n$, $i \in N$ tal que $N_i^\sigma = S$ (y por tanto, para una medida difusa $\nu\colon 2^N \to [0,+\infty)$, $\nu_i^\sigma = \nu_S$.).*

Definición III.6. Diremos que una medida difusa $\nu\colon 2^N \to [0,+\infty)$ es *estricta en $k \in N$* si, o bien $k = n$ o existe $\sigma \in \Sigma_n$ tal que $\nu_k^\sigma > \nu_{k+1}^\sigma$. Diremos que ν es *estricta* si es esctricta en k para todo $k \in N$.

Proposición III.7. *Sea $\nu\colon 2^N \to [0,+\infty)$ una medida difusa. Si*

$$\alpha_1,\dots,\alpha_n,\alpha_1+\beta,\dots,\alpha_n+\beta \geq 0,$$

entonces $\alpha_1\mathbf{OS}_1 + \cdots + \alpha_n\mathbf{OS}_n + \beta\mathbf{S}_\nu$ es creciente. Si ν es estricta en $k \in N$, entonces $\alpha_k + \beta \geq 0$; por lo tanto si ν es estricta,

$$\mathcal{I}(\mathbf{OS}_1,\dots,\mathbf{OS}_n,\mathbf{S}_\nu) = \\ = \{(\alpha_1,\dots,\alpha_n,\beta) \mid \alpha_1,\dots,\alpha_n,\alpha_1+\beta,\dots,\alpha_n+\beta \geq 0\}.$$

Proposición III.8. *Sea $\nu\colon 2^N \to [0,+\infty)$ una medida difusa. Tenemos*

$$\mathcal{I}(\mathbf{AM},\mathbf{S}_\nu) = \{(\alpha,\beta) \in \mathbb{R}^2 \mid \alpha,\alpha+n\beta \geq 0\}.$$

D. Generalizaciones de la integral de Sugeno y combinaciones

Definición III.9. Sea $\mathbb{U}$ un subconjunto conexo de $\mathbb{R}$ tal que $0 \in \mathbb{U}$. Una *medida $\mathbb{U}$-difusa en N* es una función $\nu\colon 2^N \to \mathbb{U}$ tal que

1) $\nu(\emptyset) = 0$ y
2) $S \subseteq T \subseteq N$ implica $\nu(S) \leq \nu(T)$.

Obviamente $\operatorname{im}\nu \subseteq [0,+\infty)$, pero necesitamos esta suposición más general en $\mathbb{U}$ para lo siguiente.

Definición III.10. Sean $\mathbb{U}$ y $\mathbb{I}$ dos subconjuntos conexos de $\mathbb{R}$ tal que $0 \in \mathbb{U} \subseteq \mathbb{I}$. Sea $\nu\colon 2^N \to \mathbb{U}$ una medida $\mathbb{U}$-difusa.

Diremos que las funciones $\mathrm{F}\colon \mathbb{I}\times\mathbb{U} \to \mathbb{I}$ y $\mathrm{G}\colon \mathbb{I}^n \to \mathbb{U}$ son *ν-admisibles* si la función $\mathbf{A}\colon \mathbb{I}^n \to \mathbb{I}$ dada por

$$\mathbf{A}(x_1,\dots,x_n) = \mathrm{G}(\mathrm{F}(x_{\sigma(1)},\nu(N_1^\sigma)),\dots,\mathrm{F}(x_{\sigma(n)},\nu(N_n^\sigma))),$$

donde $\sigma \in \mathbf{x}_{(\nearrow)}$ y $N_i^\sigma = \{\sigma(i),\dots,\sigma(n)\}$, están bien definidas (esto es, el resultado no cambia para $\sigma,\sigma' \in \mathbf{x}_{(\nearrow)}$) y son crecientes. Fijamos entonces $\mathbf{A} = \mathbf{A}(\mathrm{F},\mathrm{G},\nu)$ y la nombramos *$(\mathrm{F},\mathrm{G},\nu)$-función Sugeno-semejante*

Nos vamos a centrar en concreto en el caso $\mathbf{D}_\nu = \mathbf{A}(\Pi,\Sigma,\nu)$

Para el resto de la sección $\nu\colon 2^N \to \mathbb{R}$ es una medida difusa simétrica.

Resuta inmediato observar que $\mathbf{D}_\nu(c\,\mathbf{x}) = c\,\mathbf{D}_\nu(\mathbf{x})$ para todo $\mathbf{x} \in \mathbb{R}^n$, $c \in [0,+\infty)$, esto es, $\mathbf{D}_\nu$ es homogénea positiva.

Proposición III.11. *Las siguientes afirmaciones son equivalentes*

1) $\nu_1 + \cdots + \nu_n = 1$.
2) $\mathbf{D}_\nu$ *es idempotente.*
3) $\mathbf{D}_\nu$ *es invariante a traslación.*

Por supuesto esta situación no aparece si $\nu_1 = 1$, excepto en el caso trivial en que $\nu_2 = \cdots = \nu_n = 0$.

Proposición III.12. *Sea $M \subseteq N$ y fijemos $M' = N \setminus M$. $\sum_{i\in M} \alpha_i\mathbf{OS}_i + \beta\mathbf{D}_\nu$ es creciente si y solo si $\alpha_i + \beta\nu_i \geq 0$ si $i \in M$ y $\beta\nu_i \geq 0$ si $i \in M'$.*

Corolario III.13. *$\alpha\mathbf{AM} + \beta\,\mathbf{D}_\nu$ es creciente si y solo si, o bien $\beta \geq 0$ y $\alpha + n\beta\nu_n \geq 0$, o $\beta \leq 0$ y $\alpha + n\beta\nu_1 \geq 0$.*

Proposición III.14. *Sea ν estricta. Entonces $\alpha\,\mathbf{D}_\nu + \beta\,\mathbf{S}_\nu$ es creciente si y solo si $\alpha \geq 0$ y $\alpha\nu_n + \beta \geq 0$.*

E. Capa CombPool: Combinación de funciones crecientes como función de pooling

Ahora procedemos a introducir un reemplazo para las capas de pooling máximo y media aritmética clásicas, a la que llamamos capa combinación de poolings (o, abreviada, capa CombPool). El proceso es el siguiente: inicialmente escogemos r funciones crecientes $\mathbf{A}_1, \ldots, \mathbf{A}_r$ tales que $\mathbf{A}_i : \mathbb{R}^n \to \mathbb{R}\ \forall i = 1, \ldots, r$ y generamos r coeficientes $\alpha_1, \alpha_2, \ldots, \alpha_r \in \mathbb{R}, \alpha_i \geq 0\ \forall i = 1, \ldots, r$. Al reducir los $\mathbf{x} = (x_1, \ldots, x_n)$ valores de cada ventana disjunta de tamaño $k_1 \times k_2$, con $k_1 \cdot k_2 = n$ de un canal de una imagen de características X^c, utilizamos las n funciones crecientes, generando n salidas diferentes $y_1, y_2, \ldots, y_n$. La salida producida por nuestra capa se termina de calcular mediante la siguiente combinación de funciones crecientes:

$$y = \sum_{i=1}^{r} \alpha_i^2 \cdot \mathbf{A}_i(\mathbf{x}) = \sum_{i=1}^{r} \alpha_i^2 \cdot y_i \tag{2}$$

La Fig. 1 ilustra el proceso anterior mediante un ejemplo.

Observe que, con el fin de garantizar que la función resultante sea creciente, no ponderamos cada salida y_i directamente con el valor de α_i, sino con el de su potencia al cuadrado. De este modo garantizamos que la diferenciabilidad y monotonía se conservan para todas las posibles combinaciones.

A la hora de escoger un conjunto de coeficientes existen distintas opciones, tal y como se presenta en [12]: el mismo conjunto para todas las ventanas y canales de la misma capa; un conjunto diferente para cada ventana diferente (compartido por todos los canales); un conjunto diferente por cada canal (compartido por todas las ventanas de cada canal); un conjunto diferente por cada ventana de cada canal. Aunque hemos hecho pruebas con todas las variantes, consideramos que los resultados no difieren lo suficiente como para justificar reportar todas las combinaciones posibles, y nos hemos decantado por la opción de utilizar un conjunto de coeficientes distinto para cada canal, un punto intermedio entre capacidad de representabilidad del modelo y número de parámetros adicionales.

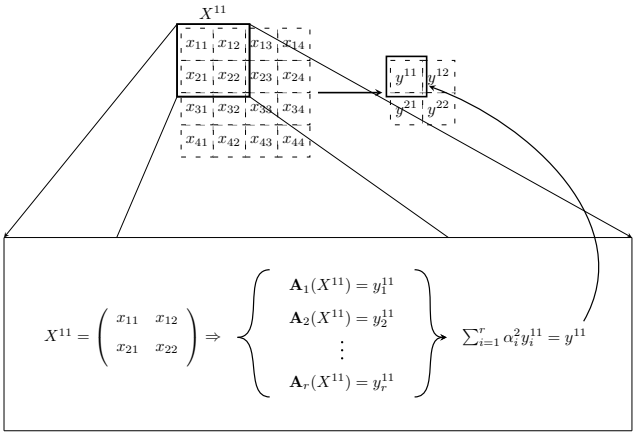


Fig. 1. Ejemplo de la combinación de funciones de pooling. Calculamos r reducciones por cada ventana, que posteriormente ponderamos mediante r parámetros $\alpha \in \mathbb{R}$.

IV. Experimentos

A. Marco experimental

En esta sección ilustramos la utilidad de la capa CombPool evaluando el rendimiento de varios modelos en los que reemplazamos sus capas de pooling por nuestra propuesta. A continuación introducimos brevemente cada una de las arquitecturas y el dataset tenido en cuenta para la evaluación.

1) Conjunto de datos: Todas las pruebas reportadas se han llevado a cabo sobre el dataset CIFAR10 [16]. Este dataset está compuesto por 50000 imágenes a color de tamaño 32x32 píxeles. Los ejemplos están completamente balanceados entre 10 clases que representan objetos del mundo real y animales.

2) Arquitecturas de CNN:

- **Arquitectura 1: LeNet-5**
  La estructura de esta red, presentada en [17] por LeCun et al. es la que se ha tomado como CNN canónica. Está compuesta por dos bloques de capas de convolución y pooling (tradicionalmente pooling media), seguido de un perceptrón de 3 capas que actúa como clasificador. Hemos fijado la función ReLU como función de activación para las capas intermedias, y la función Softmax para la capa de salida. También hemos intercalado capas de "Batch Normalization" tras cada capa de pooling [18]. Los detalles de la arquitectura se detallan en la Tabla I.
- **Architecture 2: Network in Network**
  Esta arquitectura es idéntica a la presentada en [12], y la empleamos por lo similar entre nuestra propuesta y la suya. Se basa, a su vez, en el concepto de "Network in Network" que reemplaza las capas de convolución por perceptrones multicapa y el clasificador final por una capa de Pooling Promedio Global [19].
  Adicionalmente, esta variante del modelo incluye "supervisión de capas ocultas" [20], una estrategia que intercala varios clasificadores en distintos puntos de la red para reforzar el flujo de los gradientes durante el proceso de entrenamiento y mitigar el problema de "fuga de gradiente" que tienden a sufrir los modelos profundos. Los detalles de la arquitectura pueden consultarse en la Tabla II.
- **Architecture 3: DenseNet**
  La última red que hemos empleado es una adaptación directa de la arquitectura DenseNet presentada en [6]. Las DenseNet iteran sobre el concepto de conexión identidad que las arquitecturas ResNet incluyen como método para evitar problemas de "fuga de gradiente" [3], conectando la salida de una capa con la entrada de todas las siguientes.
  Una DenseNet está formada por una serie de "bloques densos", formados por repeticiones de capas de Batch Normalization, ReLU, Convolución y, opcionalmente, Dropout [21]. Los autores se refieren mediante el factor k al número de imágenes de características que cada capa de convolución añade al conocimiento global de la red. Como medida para evitar que los filtros de capas

siguientes tengan que ser cada vez más grandes, se anteponen convoluciones 1×1 para reducir el número de canales de la entrada de la siguiente convolución a un factor fijo.

Las capas de pooling se sitúan entre bloques densos para reducir la dimensionalidad de las características extraídas. Además, tras estas capas de pooling se pueden introducir nuevamente convoluciones 1×1 para reducir el número de canales del conocimiento global de la red G por un factor θ. La Tabla III detalla la arquitectura.

TABLE I
DESCRIPCIÓN DE LA ARQUITECTURA 1.

Layer type	Output size	Kernel size
Conv1	$32 \times 32 \times 64$	3×3
ReLU	$32 \times 32 \times 64$	-
Pool1	$16 \times 16 \times 64$	2×2
BatchNorm1	$16 \times 16 \times 64$	-
Conv2	$16 \times 16 \times 64$	3×3
ReLU	$16 \times 16 \times 64$	-
Pool2	$8 \times 8 \times 64$	2×2
BatchNorm2	$8 \times 8 \times 64$	-
Flatten	4096	-
MLP1	384	-
MLP2	192	-
MLP3	10	-

B. Estudio experimental

Hemos probado a reemplazar las funciones de pooling por las funciones individuales estudiadas previamente así como por capas CombPool que combinan dichas funciones. Además del pooling media y máximo, hemos considerado el mínimo, la mediana, la integral de Sugeno y la generalización $\mathbf{D}_\nu$. La tasa de acierto media sobre 5 ejecuciones independientes de cada modelo se reporta en las Tablas IV y V.

Es interesante observar que los mejores resultados se obtienen al utilizar la función $\mathbf{D}_\nu$. El mejor global para todas las arquitecturas utiliza dicha función y, para las dos últimas, todas las combinaciones que la incluyen superan la tasa de acierto media obtenida con el resto de funciones. Curiosamente, pese a que la arquitectura 1 y 3 ofrecen resultados pobres al emplear esta función de manera individual, la arquitectura 2 obtiene su mejor resultado en este caso, lo que parece indicar que la mejor función es dependiente de la arquitectura.

Si nos fijamos en la arquitectura 2, la empleada en [12], vemos que varias de las funciones que incluyen a la integral de Sugeno superan a la propuesta presentada en dicho artículo.

Otra conclusión interesante es que los modelos más complejos parecen tener una mayor capacidad de adaptación a las funciones menos comunes. En el caso de la arquitectura 1, media y máximo ofrecen un rendimiento muy superior al del resto de funciones individuales, y para que la estrategia de combinarlas surta efecto al menos una de las dos debe estar presente. Los modelos más complejos no sufren este problema, probablemente dado que su mayor número de parámetros les permite incorporar más fácilmente estas nuevas funciones.

TABLE II
DESCRIPCIÓN DE LA ARQUITECTURA 2. LA SALIDA DE LAS CAPAS "HLSUPERVISION" NO SE USAN COMO ENTRADA DE LA CAPA SIGUIENTE. ESTAS CAPAS RECIBEN SU ENTRADA A PARTIR DE LA ANTERIOR A LA CAPA "HLSUPERVISION".

Layer type	Output size	Kernel size
Conv1	$32 \times 32 \times 128$	3×3
ReLU	$32 \times 32 \times 128$	-
HLSupervision1	10	-
Conv2	$32 \times 32 \times 128$	3×3
ReLU	$32 \times 32 \times 128$	-
HLSupervision2	10	-
Mlpconv1	$32 \times 32 \times 128$	1×1
Pool1	$16 \times 16 \times 128$	3×3
Conv3	$16 \times 16 \times 192$	3×3
ReLU	$16 \times 16 \times 192$	-
HLSupervision3	10	-
Conv4	$16 \times 16 \times 192$	3×3
ReLU	$16 \times 16 \times 192$	-
HLSupervision4	10	-
Mlpconv2	$16 \times 16 \times 192$	1×1
Pool2	$8 \times 8 \times 192$	3×3
Conv5	$8 \times 8 \times 256$	3×3
ReLU	$8 \times 8 \times 256$	-
HLSupervision5	10	-
Conv6	$8 \times 8 \times 256$	3×3
ReLU	$8 \times 8 \times 256$	-
HLSupervision6	10	-
Mlpconv3	$8 \times 8 \times 256$	1×1
Mlpconv4	$8 \times 8 \times 10$	1×1
GlobalAvgPool	10	8×8

V. CONCLUSIONES Y LÍNEAS FUTURAS

En este artículo hemos presentado las condiciones necesarias para garantizar que la combinación de funciones crecientes genere una función creciente. Además, presentamos una capa de pooling alternativa, la capa CombPool que hace uso de estas combinaciones, cuya aplicabilidad hemos probado sobre varias arquitecturas de Deep Learning de distinta complejidad.

Así mismo, hemos comprobado que la aplicación de funciones no estándar en la literatura, en concreto una generalización de la integral de Sugeno mejora en general el funcionamiento de los modelos más complejos, por lo que recomendamos su aplicación en modelos profundos de muchas capas.

A futuro, nos gustaría probar la efectividad de otras funciones de la teoría de agregaciones y su efecto al combinarlas en la capa CombPool con las ya exploradas.

AGRADECIMIENTOS

Este trabajo ha sido financiado por los proyectos de investigación PID2019-108392GB-I00 (AEI/10.13039/501100011033) de la Agencia Estatal de Investigación, PC095-096 FUSIPROD y P18-FR-4961.

REFERENCIAS

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

TABLE III
DESCRIPCIÓN DE LA TERCERA ARQUITECTURA. EN NUESTROS EXPERIMENTOS FIJAMOS k A 12 Y EL FACTOR DE COMPRESIÓN θ A 0.5.

Layer type		Output size	Kernel size
Initial block	Conv	$32 \times 32 \times 2k$	3×3
	BatchNorm	$32 \times 32 \times 2k$	-
	ReLU	$32 \times 32 \times 2k$	-
Dense block 1 ($\times 16$)	BatchNorm	$32 \times 32 \times G$	-
	ReLU	$32 \times 32 \times G$	-
	Conv	$32 \times 32 \times 4k$	1×1
	BatchNorm	$32 \times 32 \times 4k$	-
	ReLU	$32 \times 32 \times 4k$	-
	Conv	$32 \times 32 \times k$	3×3
Transition block 1	BatchNorm	$32 \times 32 \times G$	
	ReLU	$32 \times 32 \times G$	
	Conv	$32 \times 32 \times G/2$	1×1
	Pool	$16 \times 16 \times G/2$	
Dense block 2 ($\times 16$)	BatchNorm	$16 \times 16 \times G$	-
	ReLU	$16 \times 16 \times G$	-
	Conv	$16 \times 16 \times 4k$	1×1
	BatchNorm	$16 \times 16 \times 4k$	-
	ReLU	$16 \times 16 \times 4k$	-
	Conv	$16 \times 16 \times k$	3×3
Transition block 2	BatchNorm	$16 \times 16 \times G$	
	ReLU	$16 \times 16 \times G$	
	Conv	$16 \times 16 \times G/2$	1×1
	Pool	$8 \times 8 \times G/2$	
Dense block 3 ($\times 16$)	BatchNorm	$8 \times 8 \times G$	-
	ReLU	$8 \times 8 \times G$	-
	Conv	$8 \times 8 \times 4k$	1×1
	BatchNorm	$8 \times 8 \times 4k$	-
	ReLU	$8 \times 8 \times 4k$	-
	Conv	$8 \times 8 \times k$	3×3
GlobalAvgPool		G	8×8
MLP		10	-

TABLE IV
TASA DE ACIERTO PARA LAS ARQUITECTURAS USANDO FUNCIONES INDIVIDUALES.

	A1	A2	A3
AM	77.08	87.65	**89.25**
Max	**77.39**	87.85	87.99
Min	70.24	87.61	88.28
Median	70.62	87.07	88.76
$\mathbf{S}_\nu$	72.47	86.79	88.97
$\mathbf{D}_\nu$	73.42	**88.70**	87.20

TABLE V
TASA DE ACIERTO PARA MODELOS QUE UTILIZAN CAPAS COMBPOOL.

	A1	A2	A3
Min + Max	**77.02**	87.42	88.55
Min + Max + Median	76.91	**87.43**	**89.77**
AM + Min	75.04	87.23	89.48
AM + Max	**77.23**	**87.78**	86.99
AM + Min + Max	77.17	87.25	88.52
AM + Min + Max + Median	77.04	87.57	**89.83**
$\mathbf{S}_\nu$ + Min	72.41	87.46	88.58
$\mathbf{S}_\nu$ + Max	77.09	87.60	89.48
$\mathbf{S}_\nu$ + Min + Max	**77.30**	87.01	89.42
$\mathbf{S}_\nu$ + Min + Max + Median	77.03	87.29	**89.66**
$\mathbf{S}_\nu$ + AM	76.93	**88.23**	86.99
$\mathbf{D}_\nu$ + Min	72.19	88.51	89.03
$\mathbf{D}_\nu$ + Max	76.80	88.20	89.58
$\mathbf{D}_\nu$ + Min + Max	**77.81**	**88.61**	89.83
$\mathbf{D}_\nu$ + Min + Max + Median	76.15	88.30	89.75
$\mathbf{D}_\nu$ + AM	76.39	88.40	**89.87**
$\mathbf{D}_\nu$ + $\mathbf{S}_\nu$	74.92	88.03	89.68

[2] S. Liu and W. Deng, "Very deep convolutional neural network based image classification using small training sample size," in *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)*, 2015, pp. 730–734.

[3] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[4] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[5] M. Tan and Q. V. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *ICML*, 2019.

[6] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.

[7] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.

[8] Y. He, X. Zhang, and J. Sun, "Channel pruning for accelerating very deep neural networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 1389–1397.

[9] M. D. Zeiler and R. Fergus, "Stochastic pooling for regularization of deep convolutional neural networks," *arXiv preprint arXiv:1301.3557*, 2013.

[10] B. Graham, "Fractional max-pooling," *arXiv preprint arXiv:1412.6071*, 2014.

[11] J. Forcén, M. Pagola, E. Barrenechea, and H. Bustince, "Learning ordered pooling weights in image classification," *Neurocomputing*, vol. 411, pp. 45–53, 2020.

[12] C. Lee, P. Gallagher, and Z. Tu, "Generalizing pooling functions in cnns: Mixed, gated, and tree," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 863–875, 2018.

[13] O. Abdel-Hamid, L. Deng, and D. Yu, "Exploring convolutional neural network structures and optimization techniques for speech recognition," in *INTERSPEECH*, 2013.

[14] S. Ruder, "An overview of gradient descent optimization algorithms," 09 2016.

[15] F. Bardozzo, B. De La Osa, L. Horansk, J. Fumanal-Idocin, M. delli Priscoli, L. Troiano, R. Tagliaferri, J. Fernandez, and H. Bustince, "Sugeno integral generalization applied to improve adaptive image binarization," *Information Fusion*, vol. 68.

[16] A. Krizhevsky, G. Hinton *et al.*, "Learning multiple layers of features from tiny images," 2009.

[17] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[18] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proceedings of the 32nd International Conference on International Conference on Machine Learning*, ser. ICML'15, vol. 37. JMLR.org, 2015, pp. 448–456.

[19] M. Lin, Q. Chen, and S. Yan, "Network in network," *CoRR*, vol. abs/1312.4400, 2014.

[20] C.-Y. Lee, S. Xie, P. Gallagher, Z. Zhang, and Z. Tu, "Deeply-supervised nets," in *Artificial intelligence and statistics*, 2015, pp. 562–570.

[21] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

Evaluation of the Transformer Architecture for Univariate Time Series Forecastingn*

*Note: The full contents of this paper have been published in the volume Lecture Notes in Artificial Intelligence 12882 (LNAI 12882)

Pedro Lara-Benítez
Division of Computer Science
University of Sevilla
Seville, Spain
plbenitez@us.es

Luis Gallego-Ledesma
Division of Computer Science
University of Sevilla
Seville, Spain

Manuel Carranza-García
Division of Computer Science
University of Sevilla
Seville, Spain

José M. Luna-Romera
Division of Computer Science
University of Sevilla
Seville, Spain

***Abstract*—The attention-based Transformer architecture is earning increasing popularity for many machine learning tasks. In this study, we aim to explore the suitability of Transformers for time series forecasting, which is a crucial problem in different domains. We perform an extensive experimental study of the Transformer with different architecture and hyper-parameter configurations over 12 datasets with more than 50,000 time series. The forecasting accuracy and computational efficiency of Transformers are compared with state-of-the-art deep learning networks such as LSTM and CNN. The obtained results demonstrate that Transformers can outperform traditional recurrent or convolutional models due to their capacity to capture long-term dependencies, obtaining the most accurate forecasts in five out of twelve datasets. However, Transformers are generally more difficult to parametrize and show higher variability of results. In terms of efficiency, Transformer models proved to be less competitive in inference time and similar to the LSTM in training time.**

***Index Terms*—Time series, Forecasting, Attention, Transformers, Deep learning.**

DOI: https://doi.org/10.1007/978-3-030-85713-4_11

Electricity Consumption Time Series Forecasting Using Temporal Convolutional Networks*

*Note: The full contents of this paper have been published in the volume Lecture Notes in Artificial Intelligence 12882 (LNAI 12882)

J. F. Torres
Data Science & Big Data Lab
Pablo de Olavide University
Seville, Spain
jftormal@upo.es

M. J. Jiménez-Navarro
Data Science & Big Data Lab
Pablo de Olavide University
Seville, Spain
mjjimnav@upo.es

F. Martínez-Álvarez
Data Science & Big Data Lab
Pablo de Olavide University
Seville, Spain
fmaralv@upo.es

A. Troncoso
Data Science & Big Data Lab
Pablo de Olavide University
Seville, Spain
atrolor@upo.es

***Abstract*—Forecasting electricity demand is crucial for the management of smart grids to ensure a secure, reliable and sustainable supply. Recently, a variant of convolutional neural networks, called temporal convolutional networks, has emerged for data sequence, competing directly with deep recurrent neural networks in terms of execution time and memory requirements. In this work, we propose a deep temporal convolutional network to predict time series, namely, the electricity consumption with a 4-hour forecast horizon. Results using nine and a half years of Spanish electricity load, with a 10-minute sampling rate, are reported and discussed. In addition, the performance of the proposed model is compared with linear regression, decision trees, gradient boosted trees, random forests, deep feed forward neural networks that use different techniques to find the optimal hyperparameters and a deep Long Short-Term Memory network. The proposed model reaches competitive results in terms of accuracy, with the smallest error verging on 1%.**

***Index Terms*—Deep learning, Temporal convolutional networks, Time series forecasting, Electricity consumption.**

DOI: https://doi.org/10.1007/978-3-030-85713-4_21

ReLU-based Activations: Analysis and Experimental Study for Deep Learning*

*Note: The full contents of this paper have been published in the volume Lecture Notes in Artificial Intelligence 12882 (LNAI 12882)

Víctor Manuel Vargas
Universidad de Córdoba
Córdoba, Spain
vvargas@uco.es

David Guijo-Rubio
Universidad de Córdoba
Córdoba, Spain
dguijo@uco.es

Pedro Antonio Gutiérrez
Universidad de Córdoba
Córdoba, Spain
pagutierrez@uco.es

César Hervás-Martínez
Universidad de Córdoba
Córdoba, Spain
chervas@uco.es

***Abstract*—Activation functions are used in neural networks as a tool to introduce non-linear transformations into the model and, thus, enhance its representation capabilities. They also determine the output range of the hidden layers and the final output. Traditionally, artificial neural networks mainly used the sigmoid activation function as the depth of the network was limited. Nevertheless, this function tends to saturate the gradients when the number of hidden layers increases. For that reason, in the last years, most of the works published related to deep learning and convolutional networks use the Rectified Linear Unit (ReLU), given that it provides good convergence properties and speeds up the training process thanks to the simplicity of its derivative. However, this function has some known drawbacks that gave rise to new proposals of alternatives activation functions based on ReLU. In this work, we describe, analyse and compare different recently proposed alternatives to test whether these functions improve the performance of deep learning models regarding the standard ReLU.**

***Index Terms*—Activation functions, deep learning, convolutional neural networks.**

DOI: https://doi.org/10.1007/978-3-030-85713-4_4

Clustering: Un paquete R para facilitar el análisis de algoritmos de agrupamiento.

1st Luis Alfonso Pérez Martos
Departamento de Informática
Universidad de Jaén, España
lapm0001@gmail.com

2nd Pedro González
DaSCI Andalusian Research Institute
Universidad de Jaén, España
pglez@ujaen

3rd Cristóbal J. Carmona
DaSCI Andalusian Research Institute
Universidad de Jaén, España
ccarmona@ujaen.es

***Abstract*—El agrupamiento es una técnica de aprendizaje no supervisado frecuentemente estudiada y aplicada cuyo objetivo es la división de los datos en grupos de objetos similares. Es bastante común entre los investigadores, ya que permite extraer conocimientos de forma rápida y sencilla. Además, su uso es adecuado para la clasificación automática de datos con el fin de revelar cierta similitud entre ellos. En este trabajo presentamos el paquete `Clustering` desarrollado íntegramente en R, que contiene un conjunto de algoritmos de agrupamiento y que busca dos objetivos: primero, agrupar los datos de forma homogénea estableciendo diferencias entre grupos; y segundo, generar un ranking de algoritmos y atributos a partir del conjunto de datos. Este paquete incorpora una GUI que facilita su uso.**

***Index Terms*—Técnicas de aprendizaje no supervisado, Agrupamiento, R, Clustering.**

I. Introducción

Explorar las propiedades de la información para generar grupos es una técnica de aprendizaje no supervisado conocida como agrupamiento [1] [2]. Esta técnica es un modelo de datos conciso en el que un conjunto de datos debe ser particionado e introducido en grupos. Estos grupos deben cumplir dos condiciones: que los grupos sean lo más dispares posible y que los elementos contenidos sean lo más parecidos. Por regla general los algoritmos de agrupamiento se basan en la optimización de una función objeto, que suele ser la suma ponderada de las distancias a los centros. En la literatura podemos agrupar los datos de múltiples formas, entre las que destacamos [3]: particionales, jerárquicos, basados en densidad, en cuadrículas y en modelos. Uno de los algoritmos más conocidos que afronta el problema del agrupamiento es el k-means [4].

Normalmente, la tarea de comparar algoritmos de agrupamientos es tediosa, ya que debe realizarse manualmente. Esto requiere tiempo y en algunos casos podemos tener problemas en la transmisión de los resultados. Por tanto, para evaluar la distribución de los datos en grupos, es necesario indicar una variable categórica, por lo que la elección de una u otra variable de un conjunto de datos puede variar los resultados de la evaluación.
En este artículo presentamos el paquete `Clustering` [1]. Se trata de un paquete que permite comparar múltiples algoritmos de agrupamiento de forma simultánea y evaluar la precisión de sus resultados. El objetivo es permitir la evaluación de un conjunto de datos para determinar qué atributos ofrecen los mejores resultados para el agrupamiento. De esta forma, evaluamos los grupos creados, su distribucción y la categorización de los datos. La estructura de esta contribución es la siguiente. En primer lugar, en la sección II se presenta el concepto de agrupamiento y sus tipos. En la sección III tenemos la definición de las medidas de validación para evaluar la distribución de los datos en los grupos. Finalmente, la sección IV describe la estructura del paquete y presenta un ejemplo completo del funcionamiento del paquete.

[1]El paquete está disponible para su descarga en el siguiente https://CRAN.R-project.org/package=Clustering

II. Agrupamiento

El análisis de los grupos es un método de aprendizaje no supervisado que constituye la piedra angular de un proceso de análisis inteligente de datos. Se utiliza para la exploración de las interrelaciones entre una colección de patrones, organizándolos en grupos homogéneos. Se denomina aprendizaje no supervisado porque, a diferencia de la clasificación (conocida como aprendizaje supervisado), no se dispone de un etiquetado a priori que pueda utilizarse en la categorización de los datos [5]. El concepto básico de agrupamiento puede expresarse de la siguiente manera: agrupar es el proceso de identificación de grupos naturales dentro de los datos multidimensionales basados en alguna medida de similitud (Euclidean, Manhattan, etc.) [6]. Esta es una definición básica de agrupamiento, por lo que las variaciones en la definición del problema pueden ser significativas, dependiendo sobre todo del modelo especificado. Por ejemplo, un modelo generativo debería definir la similitud basándose en un mecanismo generativo probabilístico, mientras que un enfoque basado en la distancia utilizará una función de distancia tradicional para cuantificarla. Además, los tipos de datos especificados tienen un impacto significativo en la definición del problema.

A. Tipos de Agrupamiento

Existe una gran variedad de algoritmos de agrupamiento que se pueden clasificar en: jerárquicos, particionales, basados en densidad, en cuadrículas y en modelos [7]:

- Jerárquico: crea un desglose jerárquico de los datos en un dendograma que divide recursivamente el conjunto de datos en grupos cada vez más pequeños. Algunos

algoritmos jerárquicos que pertenecen a este tipo son: CURE [8], CHAMELEON [9], y BIRCH [10]

- Particional: este algoritmo minimiza un criterio de agrupamiento dado reubicando iterativamente los puntos de datos entre los grupos hasta alcanzar una partición óptima. Entre los algoritmos de agrupamiento particional podemos encontrar: CLARANS, CLARA [11], K-prototipo [12], K-modo [13] y K-means [14].
- Basados en Densidad: grupos basados en regiones densas de objetos en el espacio de datos que están separados por regiones de baja densidad. Los algoritmos más destacados en este tipo de agrupamiento son: Dbscan [15] y Denclue [16].
- Basados en Cuadrícula: primero cuantifica el espacio del agrupamiento en un número finito de celdas y luego realiza las operaciones necesarias en el espacio cuantificado [17]. Algunos de los algoritmos de agrupamiento basados en cuadrículas más conocidos son: STING [18], Wave Cluster [19] y CLIQUE [20].
- Métodos Basados en Modelos: se basan principalmente en una distribución de probabilidad. Para poder medir la similitud se basa en los valores medios que el algoritmo intenta minimizar con la función de error cuadrado. Entre los algoritmos basados en modelos destacamos: SOM [21].

III. Medidas de validación de los grupos

Gran parte de los algoritmos de agrupamiento dependen de los parámetros de entrada y del conjunto de datos, por lo que la variación de los parámetros de entrada puede dar lugar a grupos que se desvíen del conjunto de datos. Para una correcta elección de los parámetros de entrada se han empleado unos índices de validez que suele definirse combinando cohesión y separación.

- Cohesión: mide la cercanía de los elementos dentro de los grupos.
- Separación: mide lo diferente que son dos grupos, calculando la distancia entre ellos.

Hay tres enfoques para estudiar la validez de los agrupamientos [22]. El primer criterio se basa en medidas externas, que evalúa los resultados de un algoritmo base a una estructura preestablecida que se impone a un conjunto de datos y refleja nuestra intuición sobre la estructura del grupo. El segundo criterio se basa en medidas internas en la que los resultados de un algoritmo de agrupamiento se evalúan en términos de cantidad involucrada en los vectores del conjunto de datos (por ejemplo, la matriz de proximidad). Un tercer criterio, conocido como criterio relativo, consiste en comparar los resultados de la ejecución de un algoritmo con otro utilizando diferentes parámetros.

Las métricas utilizadas para la validación de las medidas externas son:

- Entropy: evalúa la distribución de las categorías en grupos [23].

$$Entropy(j) = \sum_{j=1}^{m} \frac{|C_j|}{n} E_j \quad (1)$$

Donde C son los grupos, j es el tamaño de los grupos, n es el número de grupos, y m es el número total de puntos de datos. Para calcular la entropía de un conjunto de datos tenemos que calcular la distribución de clases de los objetos en cada grupo de la siguiente manera.

$$E_j = \sum_i p_{ij} log(p_{ij}) \quad (2)$$

Donde p_{ij} es la probabilidad de un punto en el grupo i de ser clasificado como clase j.

- Recall: indica la proximidad de los resultados de la medición al valor real [24].

$$Recall(i,j) = \frac{n_{ij}}{|C_i|} \quad (3)$$

n_{ij} es el número de objetos de la clase i que están en el grupo j y n_i es el número de objetos en el grupo i.

- Precision: se refiere a la dispersión del conjunto de valores obtenidos a partir de mediciones repetidas de una magnitud [24].

$$Precision(i,j) = \frac{n_{ij}}{|C_j|} \quad (4)$$

n_j es el número de objetos en el grupo j.

- F-measure: fusiona los conceptos de Precision y Recall de la información recuperada. Por lo tanto, calculamos la Precision y Recall del grupo para cada clase como.

$$F{-}measure(i,j) = \frac{2 * (Precision(i,j) * Recall(i,j))}{(Precision(i,j) + Recall(i,j))} \quad (5)$$

- Fowlkes-Mallows Index: es una medida de comparación de agrupamiento jerárquico. Sin embargo, se puede utilizar en agrupamiento plano ya que consiste en el cálculo de un índice B_i para cada nivel i = 2 ,..., n-1 de la jerarquía [25]. La medida B_i es fácilmente generalizable. Por lo tanto, puede decirse que Fowlkes es una medida que puede interpretarse como la media geométrica de la Precision (relación entre el número de objetos relevantes recuperados y el número total de objetos recuperados).

$$Fowlkes(i,j) = \sqrt{Precision(i,j) * Recall(i,j)} \quad (6)$$

- Variation information: la variación de la información o la distancia de la información compartida es una medida de la distancia entre dos grupos [26]. Esta medida está estrechamente relacionada con la información mutua (dependencia mutua entre las dos variables) [27].

$$VI(|C_i|, |C_j|) = 2H(C_i, C_j) - H(C_i) - H(C_j) \quad (7)$$

$H(C_i, C_j)$ es la Entropy conjunta de dos grupos, $H(C_i)$ es la Entropy de C_i y $H(C_j)$ es la Entropy de C_j.

Estas medidas de evaluación externas se pueden agrupar en familias [30]

- Conjuntos coincidentes: se utiliza para comparar dos particiones de datos. Son aquellos métodos que identifican la relación entre cada grupo detectado en C y su correspondencia natural con las clases en el resultado de

referencia definido por P (predicción del resultado del agrupamiento). Se pueden definir varias medidas para medir la similitud entre los grupos en C, obtenidos por el algoritmo de agrupamiento y los grupos de P, correspondientes a nuestro conocimiento previo (externas). Las métricas incluidas en este método son: Precision, Recall y F-measure.

- Correlación entre pares: busca medir la similitud entre dos particiones en igualdad de condiciones, como el resultado de un proceso de agrupación para un mismo conjunto, mediante dos métodos diferentes C_i y C_j. Se supone que los ejemplos que están en el mismo grupo en C_i deben estar en la misma clase en C_j, y viceversa. Destacamos como medida: Fowlkes-Mallows.
- Medidas basadas en la teoría de la información: se centra en la incertidumbre existente en la predicción de las clases naturales proporcionada por la partición C. Esta familia incluye medidas básicas como la Entropy y Variation information.

Las medidas de evaluación internas no requieren información externa, por lo que se centran en los siguientes enfoques a la hora de medir la separación entre grupos:

- Vinculación simple: mide la distancia entre los miembros más cercanos de los grupos.
- Enlace completo: mide la distancia entre los miembros más distantes.
- Comparación de los centroides: mide la distancia entre los centros de los grupos.

Como medidas de evaluación internas tenemos:

- Connectivity: esta medida refleja el grado en que los elementos colocados en el mismo grupo se consideran sus vecinos más cercanos en el espacio de datos, es decir, el grado de conexión de los grupos debe ser mínimo [27].

$$Connectivity = min_{1\leq i\leq K}\left(min_{1\leq j\leq K, i\neq j}\left(\frac{dist(C_i,C_j)}{max_{1\leq k\leq K}\{diam(C_k)\}}\right)\right) \quad (8)$$

  Donde $dist(C_i, C_j)$ es la distancia entre dos grupos y $diam(C_k)$ es el diámetro de un grupo concreto.
- Dunn: representa la relación de la menor distancia entre las observaciones que no están en el mismo grupo y la mayor distancia dentro del mismo grupo [28].

$$Dunn = min_{1\leq i\leq k}\left(min_{i+1\leq j\leq k}\left(\frac{dist(C_i,C_j)}{max_{1\leq l\leq k} diam(C_k)}\right)\right) \quad (9)$$

  Donde $dist(C_i, C_j)$ es la distancia entre los grupos C_i y C_j y $diam(C_k)$ es el diámetro del grupo C_k.
- Silhouette index: el valor de silhouette es una medida de similitud de un objeto con su propio grupo (cohesión) en comparación con otros grupos (separación) [29].

$$S = \frac{1}{N}\sum_{i=0}^{N}\frac{b_i - a_i}{max(a_i, b_i)} \quad (10)$$

  donde

$$a_i = \frac{1}{|C_j| - 1}\sum_{y\in C_j, y\neq x_i}\|y - x_i\|$$

  y

$$b_i = \min_{l\in H, l\neq j}\frac{1}{|C_l|}\sum_{y\in C_l}\|y - x_i\|$$

  con

$$x_i \in C_j, H = \{h : 1 \leq h \leq K\}$$

IV. El paquete Clustering

La idea de desarrollar este paquete surge por la necesidad de establecer un marco común, que permita validar los parámetros de entrada y elegir un correcto agrupamiento de datos a partir de un conjunto de algoritmos de agrupamiento con sus respectivas medidas de similitud.

Los principales aportes del paquete son los siguientes:

- Ejecutar de forma simultánea algoritmos de agrupamiento sin necesidad de hacerlo forma individual con la finalidad de poder comparar los resultados.
- Evaluación del conjunto de medidas de similitud implementadas para cada uno de los algoritmos.
- Posibilidad de ejecutar los algoritmos utilizando como parámetro de entrada un rango de valores para su comparación.
- Evaluar la calidad de los grupos a partir de los datos de entrada, con la posibilidad de mostrar los resultados a nivel de atributos del conjunto de datos.
- Visualización de los resultados a partir de una interfaz gráfica.

El paquete está escrito en lenguaje R y tiene la capacidad de leer datos en diferentes formatos como CSV, KEEL, ARFF (Weka) y objetos dataframe. Este paquete implementa funcionalidad no desarrollada hasta ahora. En las implementaciones existenes hasta ahora no se permite la ejecución de varios algoritmos de forma conjunta. Esta opción es muy útil para comparar los resultados de varios algoritmos. Otra opción muy útil, no encontrada hasta ahora, es la posibilidad de calcular todas las medidas de similitud implementadas para un algoritmo en la misma ejecución, por ejemplo, ejecutar el algoritmo K-means para las medidas de similitud Euclidean y Manhattan. Esta perspectiva es bastante interesante cuando se trata de saber qué medida utilizar en la ejecución del algoritmo. Para evaluar la calidad de los grupos encontramos paquetes que realizan evaluaciones externas o internas, pero no hay ninguna implementación que nos ofrezca una evaluación conjunta de ambas. Por último, encontramos que a la hora de evaluar la calidad de los grupos es necesario indicar un atributo del conjunto de datos para la ejecución. Dependiendo del atributo elegido, los resultados pueden variar, por lo que sería importante contar con métodos que nos den el resultado de ejecutar cada una de las medidas de calidad por atributo, medida de similitud, número de grupos y algoritmo. Con este paquete podemos ejecutar simultáneamente varios algoritmos para cada una de las medidas de similitud implementadas. Además, a la hora de evaluar los resultados de las ejecuciones tenemos un conjunto de medidas que se ejecutan conjuntamente, lo que hasta ahora requería realizar una ejecución por medida. Esto permitirá en el futuro incorporar rápidamente

nuevas medidas. Los resultados de las ejecuciones se muestran de forma visual.

A. Algoritmos del paquete

Los algoritmos disponibles en el paquete podemos clasificarlo de la siguiente manera:

- Agrupamiento Jerárquico: agnes, clara, daisy, diana, fanny, fuzzy_cm, fuzzy_gk, fuzzy_gg, hcluster, mona, pam, pvpick and pvclust.
- Agrupamiento Particional: gama, gmm, kmeans.
- Agrupamiento Aglomerativo: aggExCluster, apclusterK.

B. Ejecución del paquete Clustering

Para poder hacer uso del paquete, previamente es necesario instalarlo. La forma más rápida de instalar es:

```
> install.package("Clustering")
```

Hay una versión de desarrollo disponible en el repositorio de github https://github.com/laperez/Clustering.

Concluido el proceso de instalación ya podemos iniciar la ejecución del método principal del paquete que es `clustering()`. Con este método podemos comparar los algoritmos de agrupamiento incluidos en el paquete. La comparación se puede realizar por paquetes o simplemente indicando los algoritmos. Para evaluar cómo se distribuyen los datos en los grupos, se dispone de un conjunto de medidas de evaluación que devuelven valores numéricos. Una mejora incorporada en el paquete es que además de devolver el valor numérico de las medidas, puede devolver el atributo del conjunto de datos correspondiente a ese valor. Además, los algoritmos se ejecutan para todas las medidas de similitud implementadas. Toda esta funcionalidad está incorporada en el método principal. Por lo tanto, los parámetros del método principal `clustering()` son los siguientes:

- path: es la ruta donde se ubica un directorio de conjunto de datos. Sólo se permite utilizar path o df pero no ambos al mismo tiempo. Los formatos permitidos son: .dat, .csv o .arff.
- df: es una matriz de datos, un data frame o matriz de similitud.
- packages: es un array de cadenas con los paquetes a ejecutar. Los paquetes son: cluster, ClusterR, advclust, amap, apcluster, gama, pvclust. Por defecto el sistema los ejecuta todos.
- algorithms: es una array de cadenas con los algoritmos implementados por los paquetes. Los algoritmos son: apclusterK, agnes, clara, daisy, diana, fanny, fuzzy_cm, fuzzy_gg, fuzzy_gk, gama, gmm, hclust, kmeans_arma, kmeans_rcpp, mini_kmeans, mona, pam, pvclust.
- min: es un campo numérico con el número mínimo de grupos. Este campo debe rellenarse y por defecto el sistema lo inicializa a 3.
- max: es un campo numérico con el número máximo de grupos. Al igual que el campo min es obligatorio y está inicializado a 4.
- metrics: es un array de cadenas con las medidas de evaluación implementadas para evaluar la distribucción de los datos en los grupos. Las medidas implementadas son: connectivity, dunn, entropy, f_measure, fowlkes_mallows_index, precision, recall, silhouette y variation_information
- attributes: es un campo boolean para indicar si queremos mostrar además de los datos numéricos de las medidas de evaluación, el atributo del conjunto de datos con el que se corresponde.

Una vez terminada la definición de los atributos de entrada del método `clustering()`, presentamos a continuación una prueba de su ejecución. Para ello utilizaremos el data frame Basketball (conjunto de datos que contiene cinco atributos que representan las estadísticas de un conjunto de 96 jugadores de baloncesto), que contiene los siguientes atributos: asistencias por minuto, altura, tiempo jugado, edad y puntos por minuto. En los datos de salida del método `clustering()` además de indicar los valores numéricos de las medidas de evaluación vamos a mostrar los atributos del conjunto de datos. Estos atributos están numerados de izquierda a derecha en orden ascendente. La idea de hacerlo de esta forma es dar homogeneidad a los resultados, en lugar de mostrar los nombres de los atributos del conjunto de datos. En la prueba vamos a utilizar los algoritmos `gmm` y `clara` incorporados en los paquetes `ClusterR` y `cluster`. Como medidas de evaluación externa vamos a utilizar `precision` y como interna `connectivity`. Además vamos a calcular los resultados en un rango de grupos que van del 4 al 5.

```
> resultado = Clustering::clustering(df =
basketball, algorithm = c('gmm','clara'),
        min=4, max=5, metrics =
c('precision','connectivity'), attributes
                    = T);
```

Los atributos resultantes de la ejecución del método son:

- Algorithm: indica el algoritmo de agrupamiento utilizado en el tratamiento de los datos.
- Distance: es la medida de similitud empleada por el algoritmo para calcular la similitud de los datos.
- Clusters: es el número de grupos usado por el algoritmo. Usado en agrupamiento particional.
- Dataset: aparece el nombre del dataframe. En el caso de ejecutar el método `clustering()` con el atributo path, aparece el nombre de los ficheros.
- TimeExternal: tiempo en ejecutar las medidas externas.
- Metrics: cada métrica indicada en la ejecución está en columnas de forma individual.
- TimeInternal: tiempo en ejecutar las medidas internas.
- TimeExternalAttr: tiempo en ejecutar las medidas de evaluación externas por atributos del conjunto de datos.
- MetricsAttr: son las mismas que la columna Metrics, pero en lugar de mostrar valores numéricos muestra el atributo del conjunto de datos.

Tabla I: Resultado con la ejecución del método `clustering()` del paquete.

Algorithm	Distance	Clusters	Dataset	timeExternal	precision	connectivity	timeInternal	timeExternalAttr	precisionAttr	connectivityAttr	timeInternalAttr
gmm	gmm_euclidean	4	Clustering::basketball	0.0030	0.1822	34.09	0.0040	5	2	1	1
gmm	gmm_euclidean	4	Clustering::basketball	0.0040	0.1113	34.09	0.0040	2	4	2	5
gmm	gmm_euclidean	4	Clustering::basketball	0.0050	0.0000	34.09	0.0040	3	1	3	2
gmm	gmm_euclidean	4	Clustering::basketball	0.0060	0.0000	34.09	0.0050	1	3	4	4
gmm	gmm_euclidean	4	Clustering::basketball	0.0109	0.0000	34.09	0.0100	4	5	5	3
gmm	gmm_euclidean	5	Clustering::basketball	0.0040	0.1637	42.08	0.0040	5	2	1	1
gmm	gmm_euclidean	5	Clustering::basketball	0.0060	0.1109	42.08	0.0040	1	4	2	5
gmm	gmm_euclidean	5	Clustering::basketball	0.0070	0.0000	42.08	0.0040	3	1	3	2
gmm	gmm_euclidean	5	Clustering::basketball	0.0070	0.0000	42.08	0.0040	2	3	4	3
gmm	gmm_euclidean	5	Clustering::basketball	0.0070	0.0000	42.08	0.0050	4	5	5	4
gmm	gmm_manhattan	4	Clustering::basketball	0.0020	0.1822	35.59	0.0040	5	2	1	1
gmm	gmm_manhattan	4	Clustering::basketball	0.0030	0.1113	35.59	0.0040	1	4	2	3
gmm	gmm_manhattan	4	Clustering::basketball	0.0030	0.0000	35.59	0.0050	2	1	3	4
gmm	gmm_manhattan	4	Clustering::basketball	0.0030	0.0000	35.59	0.0050	4	3	4	2
gmm	gmm_manhattan	4	Clustering::basketball	0.0040	0.0000	35.59	0.0050	3	5	5	5
gmm	gmm_manhattan	5	Clustering::basketball	0.0020	0.1670	46.83	0.0040	5	2	1	1
gmm	gmm_manhattan	5	Clustering::basketball	0.0020	0.1114	46.83	0.0040	2	4	2	3
gmm	gmm_manhattan	5	Clustering::basketball	0.0030	0.0000	46.83	0.0048	4	1	3	4
gmm	gmm_manhattan	5	Clustering::basketball	0.0030	0.0000	46.83	0.0050	1	3	4	2
gmm	gmm_manhattan	5	Clustering::basketball	0.0040	0.0000	46.83	0.0050	3	5	5	5
clara	clara_euclidean	4	Clustering::basketball	0.0020	0.1520	48.76	0.0040	5	2	1	2
clara	clara_euclidean	4	Clustering::basketball	0.0030	0.1186	48.76	0.0050	2	4	2	1
clara	clara_euclidean	4	Clustering::basketball	0.0040	0.0000	48.76	0.0050	4	1	3	3
clara	clara_euclidean	4	Clustering::basketball	0.0040	0.0000	48.76	0.0050	1	3	4	4
clara	clara_euclidean	4	Clustering::basketball	0.0050	0.0000	48.76	0.0100	3	5	5	5
clara	clara_euclidean	5	Clustering::basketball	0.0020	0.1657	56.52	0.0040	4	2	1	1
clara	clara_euclidean	5	Clustering::basketball	0.0020	0.1041	56.52	0.0040	1	4	2	2
clara	clara_euclidean	5	Clustering::basketball	0.0030	0.0000	56.52	0.0040	5	1	3	3
clara	clara_euclidean	5	Clustering::basketball	0.0040	0.0000	56.52	0.0040	2	3	4	4
clara	clara_euclidean	5	Clustering::basketball	0.0040	0.0000	56.52	0.0040	3	5	5	5
clara	clara_manhattan	4	Clustering::basketball	0.0020	0.1709	62.15	0.0040	3	2	1	1
clara	clara_manhattan	4	Clustering::basketball	0.0020	0.0872	62.15	0.0040	1	4	2	5
clara	clara_manhattan	4	Clustering::basketball	0.0030	0.0009	62.15	0.0040	4	3	3	2
clara	clara_manhattan	4	Clustering::basketball	0.0040	0.0000	62.15	0.0040	2	1	4	3
clara	clara_manhattan	4	Clustering::basketball	0.0040	0.0000	62.15	0.0050	5	5	5	4
clara	clara_manhattan	5	Clustering::basketball	0.0020	0.1976	93.70	0.0040	5	2	1	1
clara	clara_manhattan	5	Clustering::basketball	0.0020	0.0955	93.70	0.0040	1	4	2	5
clara	clara_manhattan	5	Clustering::basketball	0.0040	0.0011	93.70	0.0040	3	3	3	2
clara	clara_manhattan	5	Clustering::basketball	0.0040	0.0000	93.70	0.0040	2	1	4	3
clara	clara_manhattan	5	Clustering::basketball	0.0040	0.0000	93.70	0.0050	4	5	5	4

- TimeInternalAttr: tiempo en ejecutar las medidas de evaluación internas por atributos del conjunto de datos.

En la Tabla I tenemos los resultados de la ejecución del método `clustering()`. Las columnas que finalizan en Attr hacen referencia a los atributos del conjunto de datos. En los resultados podemos a preciar que para el conjunto de datos Baskteball el algoritmo que mejor resultado tiene en `Precision` es `clara` ya que es el que tiene el valor más próximo a uno. Además en vista de los resultados podemos afirmar que para dicha métrica los mejores resultados se obtienen con cinco grupos y empleando la medida de similitud Manhattan. Además podemos ver que en la columna precisionAttr el atributo del conjunto de datos con el que se obtiene el mejor valor es dos que se corresponde con la altura de los jugadores. Este mismo proceso que hemos realizado para las medidas externas se transpola a las medidas internas.

A continuación detallamos el resto de métodos que componen el paquete:

- `best_ranked_external_metrics()`: método utilizado para seleccionar entre los atributos del conjunto de datos aquellos que obtienen el mejor resultado en la evaluación de la medida. Los resultados están disponible en la Tabla II.

Tabla II: Resultado del método `best_ranked_external_metrics()`.

Algorithm	Distance	Clusters	Dataset	timeExternal	precision	timeExternalAttr	precisionAttr
gmm	gmm_euclidean	4	Clustering::basketball	0.003	0.1822	5	2
gmm	gmm_euclidean	5	Clustering::basketball	0.004	0.1637	5	2
gmm	gmm_manhattan	4	Clustering::basketball	0.002	0.1822	5	2
gmm	gmm_manhattan	5	Clustering::basketball	0.002	0.1670	5	2
clara	clara_euclidean	4	Clustering::basketball	0.002	0.1520	5	2
clara	clara_euclidean	5	Clustering::basketball	0.002	0.1657	4	2
clara	clara_manhattan	4	Clustering::basketball	0.002	0.1709	3	2
clara	clara_manhattan	5	Clustering::basketball	0.002	0.1976	5	2

- `evaluate_best_validation_external_by_metrics()`: con este método apreciamos que la elección de una medida de similitud influye en los resultados tal y como aparece en la Tabla III. El funcionamiento de este método es filtrar los valores de `best_ranked_external_metrics()` y agruparlos por algoritmo y medida de similitud.

Tabla III: Método `evaluate_best_validation_external_by_metrics()`.

Algorithm	Distance	timeExternal	precision	timeExternalAttr	precisionAttr
clara	clara_euclidean	0.002	0.1657	5	2
clara	clara_manhattan	0.002	0.1976	5	2
gmm	gmm_euclidean	0.004	0.1822	5	2
gmm	gmm_manhattan	0.002	0.1822	5	2

- `evaluate_validation_external_by_metrics()`: si queremos ir más allá y determinar el mejor algoritmo a partir de los atributos del conjunto de datos, podemos hacerlo haciendo uso de esta función y cuyo resultado aparece reflejado en la Tabla IV.

Tabla IV: Método `evaluate_validation_external_by_metrics()`.

Algorithm	timeExternal	precision	timeExternalAttr	precisionAttr
clara	0.002	0.1976	5	2
gmm	0.004	0.1822	5	2

- `result_external_algorithm_by_metrics()`: método que incorpora la funcionalidad de filtrar los resultados de un algoritmo de agrupamiento por número de grupos. Se puede ver el resultado del funcionamiento en la Tabla V.
- `plot_clustering()`: mecanismo para la representación de las medidas de evaluación de forma gráfica basado en el número de grupos. Un ejemplo donde aparece el resultado de la ejecución de este método es la Figura 1.

Tabla V: Método `result_external_algorithm_by_metrics()`.

Algorithm	Clusters	timeExternal	precision	timeExternalAttr	precisionAttr
clara	4	0.002	0.1709	5	2
clara	5	0.002	0.1976	5	2

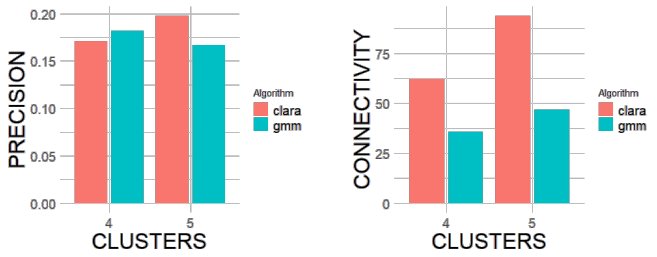


Figura. 1: Cálculo de las medidas de evaluación interna y externa para el conjunto de datos.

Estos son los métodos empleados para evaluar las métricas de evaluación externas. Además de los métodos presentados para las métricas externas el paquete dispone de métodos similares para las internas. Por último, es importante destacar que el paquete cuenta con una amplia variedad de métodos para facilitar la ordenación, filtrado y exportación de los resultados. Estas operaciones se pueden ejecutar de forma gráfica a través de la interfaz de usuario disponible en el paquete ejecutando el método `appClustering()`.

V. Conclusiones

En este trabajo hemos introducido el paquete `Clustering`. El paquete incluye una interfaz gráfica que permite al usuario ejecutarlo de forma sencilla sin necesidad de conocer los parámetros. Además contribuimos a elegir qué variable de un conjunto de datos es la que obtiene el mejor valor cuando evaluamos los grupos, así como a aplicar criterios relativos que nos permiten comparar los resultados de un algoritmo con otro utilizando diferentes parámetros. Como trabajo futuro, nos planteamos continuar el desarrollo del paquete con la inclusión de nuevos algoritmos y funcionalidades y la mejora de la interfaz de usuario.

Referencias

[1] Mann, A., & Kaur, N. (2013). Paper on Clustering Techniques By Amandeep Kaur Mann & Navneet.
[2] Steinbach, Michael & Karypis, George & Kumar, Vipin. (2000). A Comparison of Document Clustering Techniques. Proceedings of the International KDD Workshop on Text Mining.
[3] Popat, S. K., & Emmanuel, M. (2014). Review and comparative study of clustering techniques. International journal of computer science and information technologies, 5(1), 805-812.
[4] MacQueen, J.B. (1967). Some methods for classification and analysis of multivariate observations.
[5] Kotsiantis, Sotiris & Pintelas, P. (2004). Recent advances in clustering: A brief survey. WSEAS Transactions on Information Science and Applications. 1. 73-81.
[6] Omran, Mahamed & Engelbrecht, Andries & Salman, Ayed. (2007). An overview of clustering methods. Intell. Data Anal.. 11. 583-605. 10.3233/IDA-2007-11602.
[7] Patibandla, R. & Veeranjaneyulu, N.. (2018). Survey on Clustering Algorithms for Unstructured Data. 10.1007/978-981-10-7566-7-41.
[8] Sudipto Guha, Rajeev Rastogi, and Kyuseok Shim. 1998. CURE: an efficient clustering algorithm for large databases. SIGMOD Rec. 27, 2 (June 1998), 73–84.
[9] Guo, Dongwei & Zhao, Jingjing & Liu, Jici. (2019). Research and Application of Improved CHAMELEON Algorithm Based on Condensed Hierarchical Clustering Method. 14-18. 10.1145/3375998.3376016.
[10] Tian Zhang, Raghu Ramakrishnan, and Miron Livny. 1996. BIRCH: an efficient data clustering method for very large databases. SIGMOD Rec. 25, 2 (June 1996), 103–114.
[11] Ramprasanth, H., and A. Devi. (2019). "Outlier Analysis of Medical Dataset Using Clustering Algorithms." J. Anal. Comput 15: 1-9.
[12] Pitchai, Sumitra. (2018). LION OPTIMIZATION ALGORITHM USING DATA MINING CLASSIFICATION AND CLUSTERING MODELS. 6. 219 - 226.
[13] Huang, J. (1997). A Fast Clustering Algorithm to Cluster Very Large Categorical Data Sets in Data Mining. DMKD.
[14] Kushwaha, Mohit & Yadav, Himanshu & Agrawal, Chetan. (2020). A Review on Enhancement to Standard K-Means Clustering. 10.1007/978-981-15-2071-6-26.
[15] Shan Hu. 2019. Indoor Location Method Based on Data Mining. In Proceedings of the 2019 5th International Conference on Systems, Control and Communications (ICSCC 2019). Association for Computing Machinery, New York, NY, USA, 11–15.
[16] Khader, Mariam & Al-Naymat, Ghazi. (2019). An overview of various enhancements of DENCLUE algorithm. 1-7. 10.1145/3368691.3368724.
[17] Dang, Shilpa. (2011). A REVIEW OF CLUSTERING TECHIQUES IN VARIOUS APPLICATIONS FOR EFFECTIVE DATA MINING. International Journal of Research in IT & Management 2231-4434. 1. 50-66.
[18] MR, ILANGO & MOHAN, Dr. (2010). A Survey of Grid Based Clustering Algorithms. International Journal of Engineering Science and Technology. 2.
[19] Xuecheng, L. Y. L. (2010). APPLYING WAVE CLUSTER ALGORITHM IN INTRUSION DETECTION [J]. Computer Applications and Software, 6.
[20] Rani, P. (2017). A Survey on STING and CLIQUE Grid Based Clustering Methods. International Journal of Advanced Research in Computer Science, 8(5).
[21] Thalamuthu, A., Mukhopadhyay, I., Zheng, X., & Tseng, G. C. (2006). Evaluation and comparison of gene clustering methods in microarray analysis. Bioinformatics, 22(19), 2405-2412.
[22] Theodoridis, S., Koutroubas, K. (1999). Pattern Recognition, Academic Press, USA.
[23] Kim, Hyunsoo & Park, Haesun. (2007). Sparse Non-negative Matrix Factorizations via Alternating Non-negativity-constrained Least Squares for Microarray Data Analysis. Bioinformatics (Oxford, England). 23. 1495-502. 10.1093/bioinformatics/btm134.
[24] Kacprzyk, Janusz & Farhaoui, Yousef. (2019). Big Data and Smart Digital Environment. 10.1007/978-3-030-12048-1-2.
[25] Romano, Simone & the vinh, Nguyen & Bailey, James & Verspoor, Karin. (2015). Adjusting for Chance Clustering Comparison Measures. 17.
[26] He, Zengyou & Xu, Xiaofei & Deng, Shengchun. (2005). K-ANMI: A Mutual Information Based Clustering Algorithm for Categorical Data. Information Fusion. 9. 223-233.
[27] Deborah, L & Ramachandran, Baskaran & Arputharaj, Kannan. (2010). A Survey on Internal Validity Measure for Cluster Validation. International Journal of Computer Science & Engineering Survey. 1. 85-102.
[28] Ansari, Zahid & Azeem, M.F. & Ahmed, Waseem & Babu, A. (2015). Quantitative Evaluation of Performance and Validity Indices for Clustering the Web Navigational Sessions. World of Computer Science and Information Technology Journal. 1.
[29] Starczewski, Artur & Krzyżak, Adam. (2015). Performance Evaluation of the Silhouette Index. Lecture Notes in Artificial Intelligence (Subseries of Lecture Notes in Computer Science). 9120. 49-58.
[30] Palacio Niño, Julio. (2019). Evaluation Metrics for Unsupervised Learning Algorithms.

Computer Science doctoral research in Spain. A relational perspective

Adrián Arnaiz-Rodríguez*, José Miguel Ramírez-Sanz†, José Luis Garrido-Labrador‡ and Alicia Olivares-Gil§
Departamento de Ingeniería Informática
Universidad de Burgos
Burgos, España
Email: *adrianar@ubu.es, †jmrsanz@ubu.es, ‡jlgarrido@ubu.es, §aolivares@ubu.es

***Abstract*—Understanding the scientific structure of a field is pivotal for designing the strategic policy for the research development of the discipline and evaluating scientific production. The objective of this paper is to analyze and understand the structure of Computing Science as an area of knowledge in Spain. We use complex network analysis tools on the network representing the supervision and the participation in the examining committees of doctoral theses in this field. The results show a scientific network of academics characterized by a modular structure partially explained by specialization and a degree distribution compatible with a truncated power law. Also, we analyze the structure of the relationships between Artificial Intelligence and the rest of the fields, using the topic descriptors contained in each of the theses.**

***Index Terms*—Complex networks, community detection, computer science, doctoral theses, pattern recognition**

I. Introduction

Graph theory and complex networks are very useful tools for analyzing and modelling data, to discover and understand underlying structures. In this work we are particularly interested in science mapping, which is usually carried out by the formal analysis of journal citation or co-citation networks. Instead, we analyse networks built from the doctoral theses produced within a given field as entities. The relevance of this approach is twofold. On one hand it lies on the fact that the social dimension may play a more important role in the composition of the evaluation committee, given the influence of supervisors in that task. On the other, doctoral topics usually represent more robust research trends than journal and conference articles [1]. The theses considered for this study are those disserted in Spain between 1991 and 2020, on topics related to the field of computer science. The objective is to understand the scientific and social structure of this field.

II. Methogological approach

Network Science has been applied to a wide variety of scientific fields [2], [3], in particular in complex systems analysis and computational social science [4].

It is possible to analyze the static structure of a network at different scales. Several metrics capture the relative importance from the node perspective based on its relationships (centrality or prestige). There are measures to understand the overall structure of the network as a whole, such as the degree (number of relationships of the node) distribution. And it is also possible to analyze the mesoscale structure of the network; in network sciences, the community detection algorithms are used as one of the most relevant tools for this kind of analysis [5], [6].

In this study, the built network can be considered a social network since it represents the social interaction on the computer science theses evaluation process.

III. Data

The data has been retrieved from the TESEO database (https://www.educacion.gob.es/teseo) compiled by the Spanish Ministry of Education, Culture and Sports. It includes all the theses successfully defended in Spanish universities since the 70s. For each thesis, it gathers information about title, university where it was defended, author, date, supervisors, examining committee and thesis topics according to the UNESCO nomenclature for the fields of science and technology. [7].

We have retrieved more than 200,000 records from all theses registered in TESEO database. The retrieving process was the same as our previous studies in the field of organization and management of enterprises [1], [8]. Subsequently, we filtered all records to keep only those that include the code 1203 — Computer Sciences— in the UNESCO classification.

Finally, we have formalized the information gathered as a tripartite and trimodal network, that is, there are three types of nodes, with links running only between nodes of unlike types (see Fig. 1). The node types are: scholars, theses and topics. There is a link between a thesis and a scholar if the scholar has been a doctoral supervisor (S) or a member of the thesis examination committee (TEC) of the thesis dissertation. Also, there is a link between a thesis and its topics. The network includes 15845 scholar nodes, 6227 theses and 1311 topics from 29th March 1996 to 12th December 2020.

IV. Analysis

A. Degree distribution

The first useful insight over a network is usually the analysis of the degree distribution, which is the probability distribution of the degrees in the network.

It has been frequently observed in real networks, specifically on social contexts, that the degree distribution is usually heavy-tailed [9]. These types of distributions are unbounded, present extreme observations with a very high degree and have long

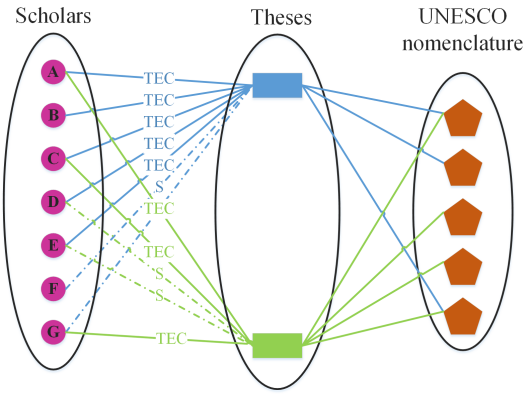


Fig. 1. Tripartite and trimodal network between scholars, theses and UNESCO.

right tails. The formation of some of these networks has been explained by a cumulative advantage or preferential attachment mechanisms [10], which follows the principle "The rich get richer", since the probability of a new node will connect to other is proportional to the other's current degree.

Exponential, Power Law, Lognormal or Truncated Power Law [11] are the most often heavy-tailed distribution found in the real world. Fitting these distributions is not trivial at all.

In this work we analyze the degree distribution on the bipartite network of scholars and theses. We only select the degree of the scholars (theses excluded); thus, we obtain the distribution of the number of theses in which an scholar has participated, along with the probability density function (PDF) using logarithmic binning and the complementary cumulative distribution function or survival function (CCDF) (see Fig. 2). The latter is shown used log-log scales on the axis and it allows us to understand at a glance the behavior and fitting of the distribution.

To obtain the distribution that best fits our data, our approach is comparing the log-likelihood ratio among the aforementioned distributions, being able to provide statistical significance for the conclusions. We use the *powerlaw* Python package [12]. Table I shows the results comparison. If the ratio is positive it means that the data are more likely to fit the row distribution. Heavy-tailed distributions are defined as those whose tails are not exponentially bounded, which have tails heavier than the exponential distribution. Since all the heavy-tailed distributions analyzed are more probable than the exponential, we can confidently assume the presence of heavy tails. We can also conclude that the Truncated Power Law is the best candidate as well.

Whereas power law suggests an unlimited linear relationship between the degree and the probability of increasing the degree, the Truncated Power Law state the power law behavior only in a range of the distribution. It usually happens that the tail decays due to an upper bounding effect as a consequence of a limited resource. In our work, the heavy-tail fit could be explained by the rare events of scholars participating in 100 theses (max. degree 103). The mechanism of creation of

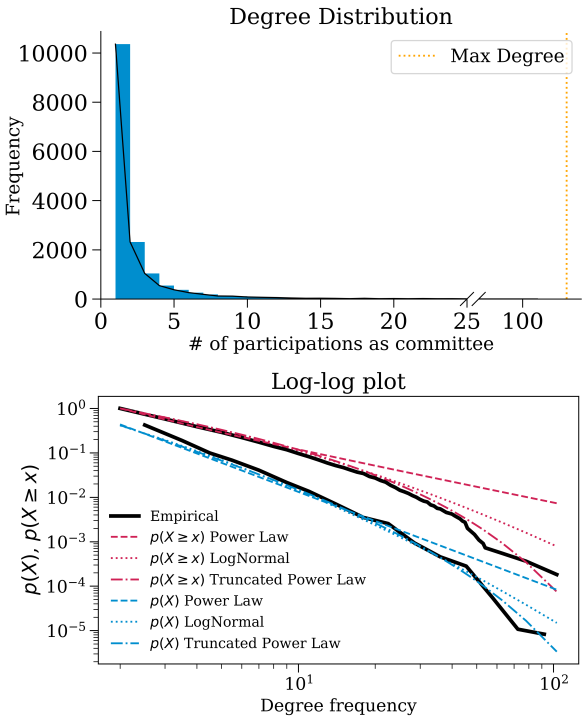


Fig. 2. Degree distribution and log-log plot

TABLE I
COMPARISON CANDIDATE DISTRIBUTIONS

R (p value) *	Exponential	Power law	Lognormal
Exponential			
Power law	9.96 (2e-23)		
Lognormal	13.06 (5e-39)	9.78 (1e-22)	
Truncated power law	13.35 (1e-40)	11.03 (0.0)	3.11 (1.8e-3)

* R is the log-likelihood ratio between pairs of distributions.

the network suggests the following: the more theses you have participated in, the more probability you have to participate in another one. However, there is a decaying process on the higher degrees since our distribution is a truncated power law. This effect could be explained by some limited resources such as age or the finite time of career. These results are similar to our previous work insights [8].

A power law distribution has the form $Y = kX^{\alpha}$. According to this result, we have computed the parameters of the truncated power law. The lower bound —the degree from it starts to behave as a power law— is 2, which shows that our distribution behaves as a power law over a wide range. In addition, α shows the strength of the linear relationship between degree and probability to be attached with a new node. The bigger α is, the more attraction power the nodes with high degree have. $\alpha = 1.8$ suggests the existence of a strong accumulation process.

B. *Communities of people and its specialization*

The following analysis is related to the social dynamics of scholars working together. Is there any group of people that

usually works together? Why do they usually work together? Is there any academic reason? What causes explain this social behavior? To answer these questions, we made a projection of the network over the scholars and performed a community detection algorithm. Then, we analyze the profile specialization of each community and figure out whether the communities are highly specialized in terms of research topics or not. We eventually find the similarities between communities to discover possible overlapping.

The first step of our approach lies in performing a simple weighting projection of the aforementioned network onto the scholars' space. We use the bipartite network of theses and scholars as the starting point of this projection. Therefore, it produces a weighted, undirected and unimodal graph. The nodes are the different researchers and they are linked on the projection with a weight proportional to the number of theses in which they have worked together. This network includes 15845 nodes —scholars— and 76199 links.

We also add two attributes to the scholars nodes in order to keep useful information on the projected network: *theses* and *profile specialization*. The former is the degree of the scholar on the original network. The latter represents a way of determining the profile specialization of a scholar conforming to the UNESCO 6-digit-level disciplines under the 1203 code —Computer Sciences—. This second attribute refers to a distribution of the 29 descriptors under 1203 UNESCO code, of the theses in which a scholar has participated. A scholar inherits the descriptors of a thesis in which she has participated as follows: the descriptors of each thesis will sum 1, thus, if a thesis has only one descriptor it will sum 1 to the descriptor value on the scholar distribution. Otherwise, if another thesis has 4 descriptors, 0.25 will be added to each current descriptor value of the scholar distribution i.e $1/n$. This process results in a vector for each academic representing the research specialization profile. Once this vector is built, it is normalized to be represented as a relative frequency vector, i.e. the values of the different UNESCO topics for each scholar will sum to 1 as if it were a probability distribution [1].

To better analyze the network, we filtered the backbone to obtain only the academics who participated in more than 10 theses. Finally, we have a projection with 449 nodes and 3891 links.

As stated before, we have identified the communities using the Louvain [13] algorithm with the objective of analyzing the social structure of Computer Science field in Spain. Communities are subsets of nodes scholars more densely connected between them than with the nodes outside the community. We find 12 communities with a modularity value of 0.579. This insight reveals the strong social connections of the scholars through modular patterns. Besides, analyzing the specialization vectors of the members belonging to a community can provide a perspective about the specialization profile of a community. Setting this profile, we are analyzing whether and how a community is specialized. This general specialization profile of each community is built using a weighted average of the specialization vector of each community member. We use the number of theses as weight. This weighted average symbolizes a consensus distribution. This vector represents a probability distribution as well. The specialization distribution of each community is shown in Fig. 3.

We also measured the degree of specialization of each community calculating the normalized entropy of the specialization vector. Entropy is not bounded between 0 and 1. It has a maximum value of $\log_2(29)$ since 29 is the number of topics under computer science. We normalize the entropy by the stated value to sort out this problem and to make our results comparable with other fields and disciplines. The higher the entropy, the less specialized is the community. Being n the number of descriptors and x_i the probability value of the topic i on the community specialization vector, the entropy of a community is:

$$S_{norm} = \frac{\sum_i^n P(x_i) \log P(x_i)}{\log n}$$

The results show that all communities are highly specialized in some subdisciplines since every community has a low entropy value. This fact could be a first evidence that scholars are organized based on the area of research.

Although every community is certainly specialized on some topics, we can not state that the social relationship of the scholars is only explained by this fact. While there is certainly specialization of the communities around the topics, our results also reveal that there are communities that are socially separated but with similar specializations at the level of scale of analysis. Therefore, some communities are overlapped, that is correlated, in terms of their specialization.

In this work, we make a further step on the analysis developed in [1] to shed light on the correlation between communities based on their research profile. To explore the relationship between communities, we perform clustering of the communities with respect to their topic distribution vectors. Consequently, we have used a hierarchical clustering algorithm, with cosine distance and complete linkage [14]. The results can be seen in Fig. 4 as well as in the background color of each plot in the Fig 3. Community 11 (blue) is certainly specialized on *Computer-Assisted Teaching*. Communities 2, 3 and 4 (green) are specialized on *Computer Software*, *Informatics* with a slight contribution on *Design and Components of Informatics Systems*. The next cluster is the one made up of communities 1 , 6 and 10 (red). Communities 1 and 10 are very close, but community 6 is a bit farther. This cluster highlights both *Informatics* and *Artificial Intelligence*, with the former a bit more present. Communities 8, 5, 0, 9 and 7 belongs to the last cluster, all of them being very similar and quite close as we can see both in the dendrogram and in the distribution plots. This cluster is focused mainly on *Artificial Intelligence*.

We also perform a t-SNE [15] projection of the specialization vectors of the communities to show the robustness of the hierarchical clustering regarding the embedding space of these vectors (Fig. 5).

Both results, entropy and clustering suggest that the communities are specialized, but there are also a big correlation

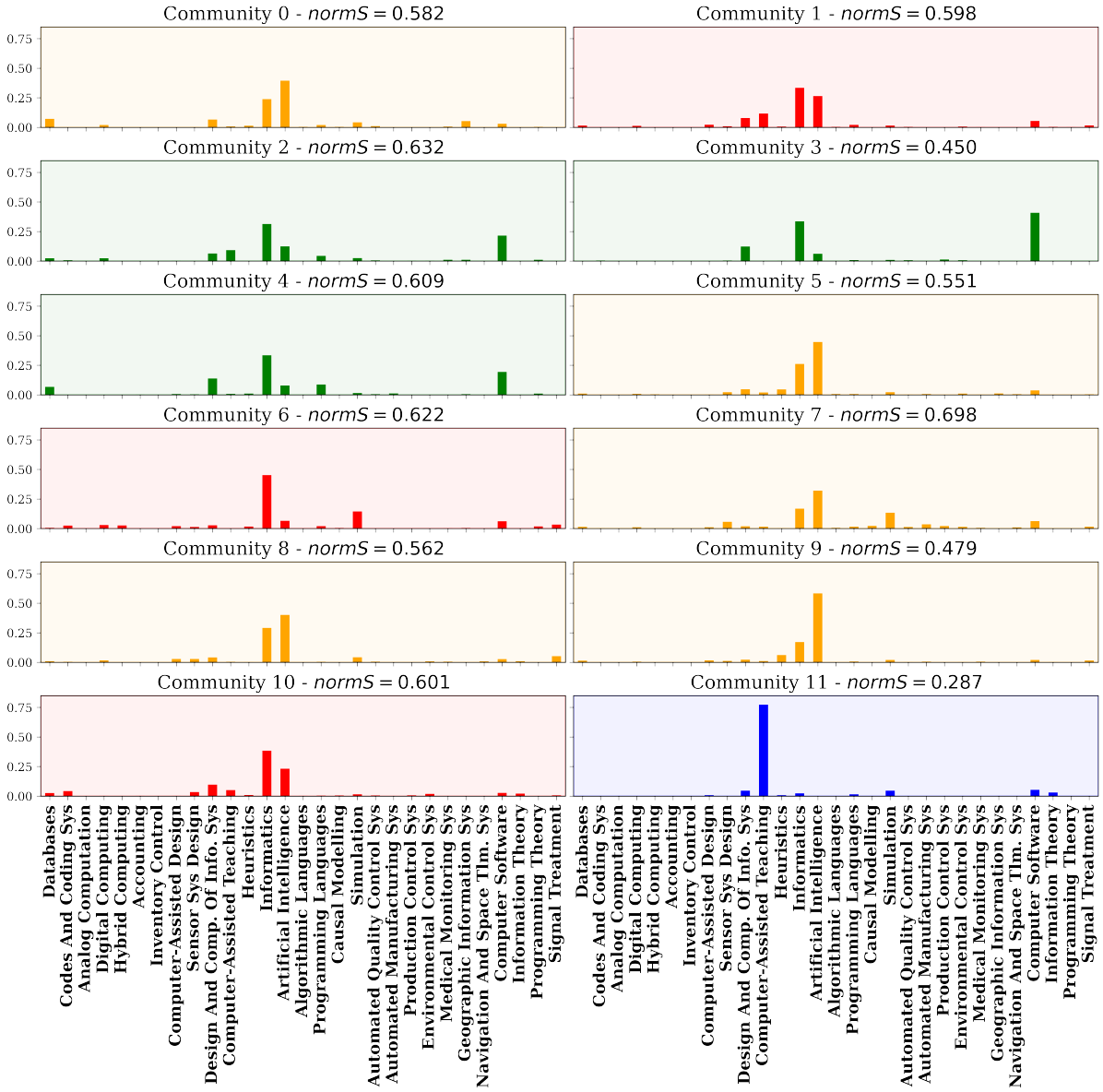


Fig. 3. The specialization profile of each community is represented by its topic distribution and normalized entropy. We also carried out a hierarchical clustering of the communities using the distribution vector of the topics of each community to discover similarities according to their specialization profile. Communities with the same graph color belong to the same cluster, which are shown on Fig. 4. Fig. 5.

between some of them in terms of the areas of research, which represents the profile specialization. The communities are certainly specialized on some topics, suggesting that the social behavior to connect with other scholars is based on the area of research. The big similarity between some communities gives us a more general view of communities organized on clusters. Each cluster contains communities with similar profile specialization, suggesting that other social effects may exist on the generation of the network, and the area of research. There are several possible causes: friendship between scholars, organizational closeness (works in similar workshops or same journals), territorial proximity, etc. But we also would like to settle a possible cause: UNESCO codes are not detailed enough. We can identify 5 different communities with a big specialization on AI. As we already know, Artificial Intelligence (AI) is currently widely spread. May the cause be that the *Artificial Intelligence* UNESCO descriptor includes lots of different subareas and it does not allow us to differentiate one subarea from another. Computer vision, natural language processing, semi-supervised algorithms, theoretical foundations, practical works are only some of the many subareas that could be redefined within AI.

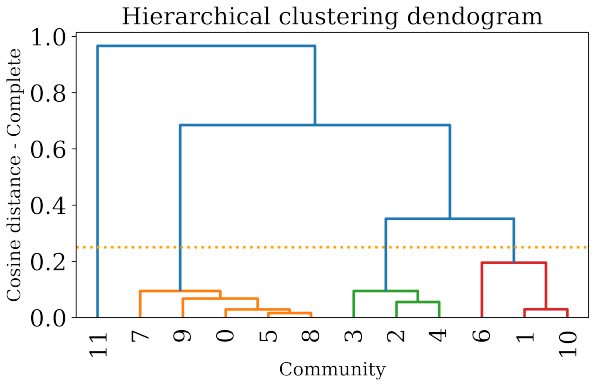


Fig. 4. Clustering of communities regarding their topic distribution vectors.

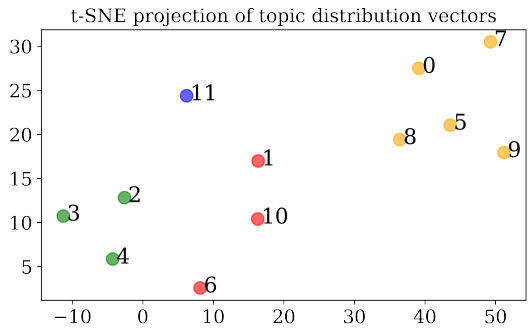


Fig. 5. t-SNE projection of topic distribution vector

C. Artificial Intelligence Ego-network

The last part of our study assesses how the subdiscipline of AI relates to other fields in Spain. Since the initial network contains all the theses related to at least one descriptor contained under 1203 code (Computer Sciences), a subsequent filtering is necessary to keep only theses related to the AI descriptor. The hyperbolic projection [16] was applied on the descriptors to transform the bipartite network into a unimodal network. The result is an ego-network with 23 nodes and 121 edges presented in the Fig. 6. Notice that all nodes correspond to an UNESCO code at a 2-digit level except for the AI central node which is at a 6-digit level. Consequently, the *Technological Sciences* node absorbs all his lower-level descriptors except for AI.

The resulting network offers a picture of the relationships between the AI subdiscipline and the other fields defined by the UNESCO thesis descriptors in Spain. We study the network by establishing various intervals for the degree of the nodes, which is an indicator of the level of relationship of each field with the central node. Given its great development in recent decades, it is expected that AI interacts, to a greater or lesser degree, with most fields of knowledge. The only 2-level descriptors that do not present any relationship are *History* and *Ethics*. From highest to lowest degree, in the first level we find *Mathematics*, *Technological Sciences*, *Physics* and *Linguistics*, followed by *Logic*, *Life Sciences* and *Medical Sciences*. The presence of *Linguistics* stands out alongside the hard sciences, probably due to the success of AI techniques in natural language processing. In the middle levels we find: *Economic Sciences* and *Psychology* in the third level; *Earth and Space Sciences* and *Astronomy and Astrophysics* in the fourth level; and *Chemistry* and *Philosophy* in the fifth level. The external position of *Chemistry* contrast with the central position of other fields of a similar nature, like *Physics* and *Medical Sciences*. In the outermost level we find *Sociology*, *Agricultural Sciences*, *Demographics*, *Pedagogy*, *Political Sciences*, *Juridical Sciences and Law*, *Geography* and *Anthropology*. The low degree of these nodes may have two explanations: there are not many cases of application of AI techniques to the field or, on the other hand, the possible applications have not yet been exploited.

A limitation of the analysis that is important to highlight is that the interaction formalized through the network does not accurately distinguish the relationship between disciplines in each case. For example, it may be a multidisciplinary interaction in which certain AI techniques are used to solve a problem in another field, it may be that other discipline developments are incorporated into the theoretical background of a subdomain of AI, or it may be relations of a different nature. The network aggregates the interactions without being able to differentiate the distinct types of multidisciplinary, interdisciplinary or transdisciplinary relationships that may take place.

V. Conclusions and Implications

First, the study shows that the degree distribution follows a truncated power law. Thus, we can conclude the power law behavior of the degree in a wide range of the distribution, decreasing at the end of the right tail due to limitation of resources such as the duration of an academic career. This type of distribution is very uneven, with academics participating very frequently in the processes and many researchers with little participation.

On the one hand, the analysis of the backbone of the projection of the theses-scholars network onto the scholars reveals that, despite the fact that the communities found are specialized, the overlapping of specialization among some of them allowed us to suggest that other underlying causes should be taken into account. However, we still wonder about the other possible causes due to the limitations of this work. Also, we could suggest some possible causes such as the limitations of the UNESCO descriptor or any of the aforementioned ones. Our results indicate that, given the growth and subspecialization of AI in recent years, a revision or expansion of the coding of the discipline may be necessary. The implications of these results are important for improving the efficiency of the search process for relevant scientific literature in a structured way.

Finally, the ego-network confirms the extent of the relationships between AI and other fields of knowledge in Spain. By presenting different circles of interaction, if is shown that there are areas more related than others. While the central nodes represent relationships already explored to a certain

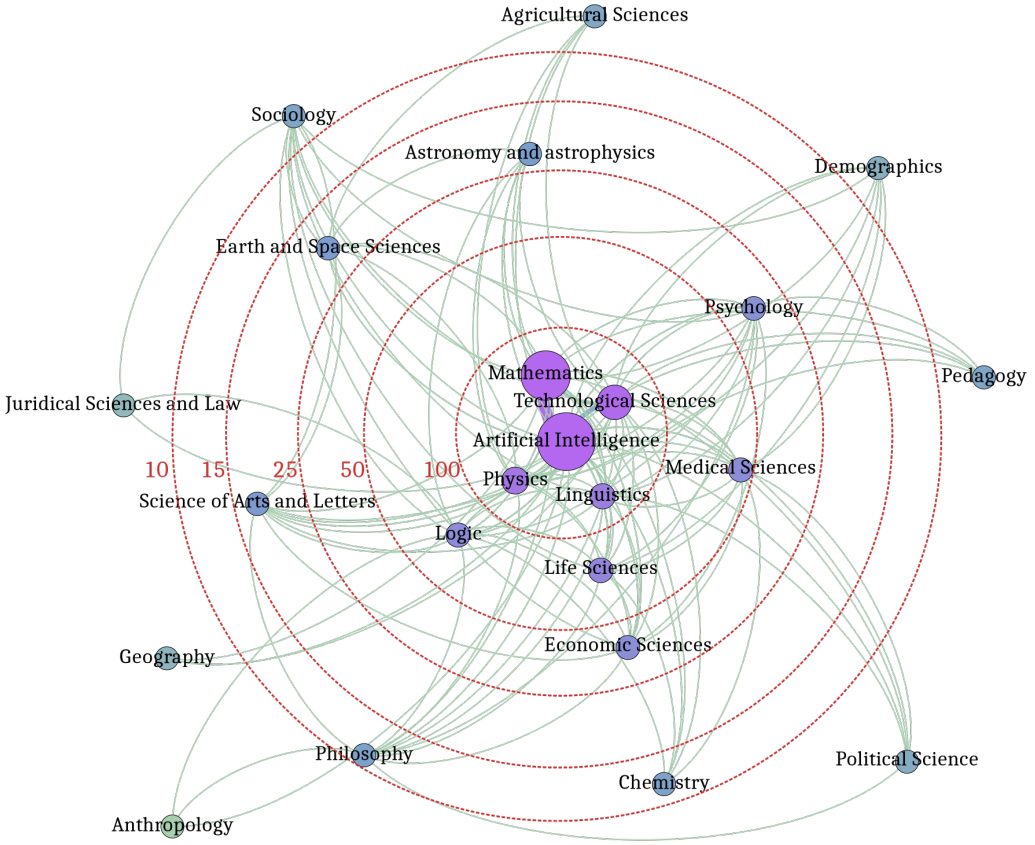


Fig. 6. Artificial Intelligence ego-network. Each node represents a UNESCO 2-level field. Red circles represent different intervals for the degree of the nodes.

point where to carry out exploitation research, the areas present in the outermost level open the door to explore innovative interdisciplinary collaborations.

VI. ACKNOWLEDGMENTS

José Luis Garrido-Labrador and Alicia Olivares-Gil were supported by the predoctoral grant (BDNS 510149) awarded by the Universidad de Burgos, Spain. In addition, the authors would like to thank Dr. García-Osorio and Dr. Galán for their advice and their invaluable help in improving the manuscript.

REFERENCES

[1] A. Olivares-Gil, A. Arnaiz-Rodríguez, J. M. Ramírez-Sanz, J. L. Garrido-Labrador, V. Ahedo, C. García-Osorio, J. I. Santos, and J. M. Galán, "Mapping the scientific structure of Organization and management of enterprises using complex networks," in *Proceedings of the 15th International Conference on Industrial Engineering and Industrial Management and XXV Congreso de Ingeniería de Organización*, J. M. Galán *et al.*, Eds. Burgos: Pressbooks, 2021.

[2] A.-L. Barabási, *Network science*. Cambridge University Press, 2016, oCLC: 910772793.

[3] M. Newman, *Networks*, 2nd ed. Oxford, New York: Oxford University Press, Jul. 2018.

[4] R. Conte *et al.*, "Manifesto of computational social science," *The European Physical Journal Special Topics*, vol. 214, no. 1, pp. 325–346, 2012.

[5] P. Bedi and C. Sharma, "Community detection in social networks," *WIREs Data Mining and Knowledge Discovery*, vol. 6, no. 3, pp. 115–135, 2016.

[6] S. Fortunato and D. Hric, "Community detection in networks: A user guide," *Physics Reports*, vol. 659, pp. 1–44, 2016, community detection in networks: A user guide.

[7] N. UNESCO, "Proposed international standard nomenclature for fields of science & technology," *Paris: United Nations Educational, Scientific and Cultural Organization*, 1988.

[8] J. L. Garrido-Labrador, J. M. Ramírez-Sanz, V. Ahedo, A. Arnaiz-Rodríguez, C. García-Osorio, J. I. Santos, and J. M. Galán, "Network analysis of co-participation in thesis examination committees in an academic field in spain," *Dirección y Organización, in press*, 2021.

[9] A. Clauset, C. R. Shalizi, and M. E. Newman, "Power-law distributions in empirical data," *SIAM review*, vol. 51, no. 4, pp. 661–703, 2009.

[10] D. D. S. Price, "A general theory of bibliometric and other cumulative advantage processes," *Journal of the American Society for Information Science*, vol. 27, no. 5, pp. 292–306, 1976.

[11] L. A. N. Amaral, A. Scala, M. Barthélémy, and H. E. Stanley, "Classes of small-world networks," *Proceedings of the National Academy of Sciences*, vol. 97, no. 21, pp. 11 149–11 152, 2000.

[12] J. Alstott, E. Bullmore, and D. Plenz, "powerlaw: a python package for analysis of heavy-tailed distributions," *PloS one*, vol. 9, no. 1, p. e85777, 2014.

[13] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of statistical mechanics: theory and experiment*, vol. 2008, no. 10, p. P10008, 2008.

[14] S. C. Johnson, "Hierarchical clustering schemes," *Psychometrika*, vol. 32, no. 3, pp. 241–254, 1967.

[15] L. van der Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research*, vol. 9, no. 86, 2008.

[16] M. E. J. Newman, "Scientific collaboration networks. ii. shortest paths, weighted networks, and centrality," *Phys. Rev. E*, vol. 64, p. 016132, Jun 2001.

Estudio comparativo de medidas de disimilitud para Clustering Multi-Instancia

Aurora Esteban
Dept. de Informática
Universidad de Córdoba
Córdoba, España
aestebant@uco.es

Amelia Zafra
Dept. de Informática
Universidad de Córdoba
Córdoba, España
azafra@uco.es

Sebastián Ventura
Dept. de Informática
Universidad de Córdoba
Córdoba, España
sventura@uco.es

***Resumen*—La tarea de *clustering* o agrupamiento consiste en encontrar la mejor agrupación de patrones en función de un criterio de similitud o disimilitud entre ellos. De esta forma, se busca que los patrones dentro de un clúster sean muy similares entre ellos y disimilares de otros clústeres. Definir el criterio de similitud entre patrones resulta así crucial en este tarea de aprendizaje. En el aprendizaje Multi-instancia, la medida de similitud o disimilitud todavía se vuelve más relevante debido a que en este aprendizaje los patrones están formados por conjuntos de instancias, donde todas ellas pueden no contribuir de la misma forma para representar al patrón. En este trabajo se lleva a cabo un estudio de siete medidas que permiten estimar la disimilitud entre patrones considerando tanto la distribución espacial como la distribución por probabilidades. Resultados experimentales utilizando 28 conjuntos de datos muestran el efecto de la medida utilizada en los resultados obtenidos por el algoritmo MIKMeans.**

***Index Terms*—Multiple Instance Learning, Clustering, Medida de disimilitud**

I. Introducción

La tarea de agrupamiento automático, en inglés *clustering*, es una de las más populares y utilizadas en el campo de la minería de datos para extraer información útil a partir de datos en crudo o sin procesar. Dicha tarea trata de agrupar los patrones en grupos o clústeres de tal manera que dentro de un grupo los objetos sean lo más similares posibles a la par que entre distintos grupos sean lo más diversos posibles. La tarea de *clustering* ha sido abordada por múltiples propuestas que pueden ser categorizadas en *clustering* particional, jerárquico, basado en densidad, basado en cuadrícula o basado el modelos [1]. En esta línea, el *clustering* ha sido aplicado a una amplia variedad de disciplinas tanto científicas como relativas a la ingeniería, entre las que se encuentran la identificación de genes, la detección de enfermedades, la visión por ordenador o el reconocimiento de patrones [1].

El aprendizaje basado en múltiples instancias, en inglés *Multiple Instance Learning* (MIL), [2] se considera una extensión del aprendizaje automático tradicional que introduce más flexibilidad a la hora de representar la información. MIL ha sido ampliamente aplicado a tareas como la categorización de textos, la anotación de los elementos presentes en una imagen o la predicción de efectos en medicamentos [3]. Actualmente, el paradigma de MIL está siendo de gran relevancia en aprendizaje no supervisado, con propuestas que lo utilizan para la estimación de la pose humana [4], [5], la identificación de estructuras en proteínas [6] o la detección en imágenes [7].

En la terminología de MIL, un objeto o patrón se llama *bolsa* y está compuesto por un número variable de *instancias*, cada una con la misma estructura de características. En aprendizaje no supervisado es difícil determinar cómo medir la distancia entre bolsas, ya que dentro de cada una puede haber instancias más informativas que otras. Por ejemplo, una bolsa que contenga la información de una imagen por segmentos, tendrá instancias menos informativas, correspondientes al fondo de la imagen, y otras más relevantes correspondientes a los elementos del primer plano. Así pues, en el *clustering* MIL es todavía más relevante determinar cómo se calcula la similitud entre bolsas. Debido a que éste es un problema complejo, pocas propuestas de *clustering* centradas en MIL pueden encontrarse en la bibliografía [5], [8]–[11].

Diferentes enfoques se han seguido en la bibliografía para evaluar cómo de diferentes son las bolsas entre sí, tanto en aprendizaje supervisado como no supervisado: distancia a nivel de bolsa, calculo del kernel a nivel de bolsa o a nivel de instancias o disimilitud entre bolsas [3]. En este artículo se trabaja desde una perspectiva de disimilitud entre conjuntos, un enfoque con potencial en MIL ya que permite eliminar restricciones matemáticas que tienen las métricas de distancia estricta, introducir conocimiento experto en el cálculo o adaptar un mismo algoritmo a diferentes dominios únicamente redefiniendo la métrica de disimilitud escogida [12]. De este modo, partiendo del algoritmo de agrupamiento particional para MIL MIKMeans [13], se analizan sus resultados empleando diferentes medidas de disimilitud que nos permitirán evaluar el efecto que tienen dichas medidas en la detección de uno o varios conceptos en la bolsa y cómo esto afecta a la calidad del agrupamiento.

El resto del trabajo se organiza como sigue. En la Sección II se hace un repaso del trabajo relacionado. En la Sección III se comentan algunos aspectos previos para poner en contexto las diferentes propuestas de disimilitud MIL, que son presentadas en la Sección IV. En la Sección V se presenta y analiza el estudio experimental. Finalmente en la Sección VI se exponen las conclusiones y se plantean futuras líneas de trabajo.

II. Trabajo Relacionado

El paradigma multi-instancia fue introducido por Dietterich et al. [2] como una forma de aprendizaje en el que las instancias de entrenamiento se agrupan en conjuntos llamados bolsas. Esta representación introduce flexibilidad en la composición de las bolsas, los tipos de distribución de datos y las relaciones entre las instancias de una bolsa. Por ello, esta formulación del problema ha atraído mucha atención de la comunidad científica dado por el aumento de la cantidad de datos disponibles [3].

Tanto la formulación original como las primeras propuestas de MIL están centradas en problemas de aprendizaje supervisado [3], con varias técnicas consolidadas para la tarea de clasificación. En aprendizaje no supervisado, específicamente en *clustering*, existen varias propuestas que adaptan algoritmos clásicos introduciendo métricas de distancia específicas para trabajar con bolsas de instancias. Henegar et al. [8] presentan un método de *clustering* particional basado en adaptar Citation-kNN al aprendizaje no supervisado, que optimiza el índice Silhouette. Kriegel et al. [9] proponen el método MIEM-Clustering, basado en la maximización de expectativas según la distribución gaussiana de bolsas. COSMIC [10] es un método de *clustering* jerárquico basado en densidad que también utiliza redes de conceptos. BAMIC [11] propone la adaptación del algoritmo K-medoids pero tratando con bolsas completas en lugar de instancias. Para medir la distancia entre bolsas, utiliza la distancia de Hausdorff. Finalmente, Zhang et al. [5] introducen M3IC-MBM, un método de agrupamiento basado en la maximización del margen entre bolsas con algunas relajaciones de las restricciones como el procedimiento cóncavo-convexo y el método del plano de corte. Además, los autores de este trabajo presentamos en [13] una serie de propuestas de *clustering* particional basados en algoritmos evolutivos con diferentes características y optimizando diferentes índices de validación internos de *clustering*: MIGKA, MIFGKA, MIGCUK y CHCMIC, así como una propuesta no evolutiva llamada MIKMeans que adapta *K*-medias a MIL utilizando la distancia de Hausdorff para medir la distancia entre bolsas.

En este trabajo se propone utilizar como base MIKMeans y estudiar cómo afecta a su rendimiento la inclusión del concepto de disimilitud entre bolsas según sus instancias frente al enfoque clásico basado en distancia.

III. Preliminares

En esta sección se hace una introducción tanto de la nomenclatura como del algoritmo utilizado en este estudio.

III-A. Nomenclatura

Los conjuntos de datos MIL se componen de N bolsas, denominadas como $U = b_1, ..., b_N$. Cada bolsa contiene un número variable de instancias m_i, de forma que cada $b_i = x_{1i}, ..., x_{mi}$ y cada instancia de cada bolsa en U es un vector de D dimensiones (los atributos de instancia del dataset MIL), de forma que $x_{ij} = (a_{1ij}, ..., a_{Dij})$. El propósito del algoritmo de clustering que se propone en el siguiente apartado será formar K clústeres $C = C_1, ..., C_K$, que se compondrán cada uno de un número variable de bolsas l, es decir, $C_i = b_{1i}, ..., b_{li}$. Además, cada C_i tiene asociado un centroide c_i, siendo el conjunto de centroides que definen el agrupamiento hecho por el algoritmo $c = c_1, ..., c_K$. Por último, para referirse al número de elementos que componen un conjunto S se utiliza el operador $|S|$.

III-B. MIKMeans

MIKMeans es una adaptación a MIL del algoritmo básico de agrupamiento particional K-medias [13]. Así, en este caso el objetivo es agrupar las bolsas del conjunto de datos en K grupos disjuntos minimizando iterativamente la distancia entre cada bolsa y el centroide del grupo asignado. El flujo del algoritmo empieza escogiendo aleatoriamente K bolsas de U y, a su vez, de cada bolsa escoge una instancia también aleatoriamente, con el fin de conformar los centroides iniciales c. A continuación, cada bolsa se asigna al centroide más cercano, construyendo así los clústeres C. Seguidamente, para cada C_k, el centroide de la siguiente iteración $c_k(next)$ se recalcula como la media de todas las instancias contenidas en todas las bolsas de C_k:

$$c_k(next) = \frac{1}{|C_k|} \sum_{b_i \in C_k} \frac{1}{|b_i|} \sum_{x_{ji} \in b_i} x_{ji} \quad (1)$$

Si los nuevos centroides coinciden con los de la iteración previa, es decir, $\forall k \in [1, K] c_k = c_k(next)$, el algoritmo ha convergido y finaliza. Si no, $c_k(next)$ sobrescribe a c, y el algoritmo repite los pasos segundo y tercero.

IV. Disimilitud en MIL

En *clustering*, la medida de distancia es ampliamente utilizada para establecer la disimilitud entre los grupos. En este trabajo se quiere estudiar cómo afecta la medida de disimilitud en la tarea de *clustering* multi-instancia. Las métricas de distancias imponen varias restricciones matemáticas: la distancia entre dos bolsas siempre debe ser positiva, y sólo será 0 si una bolsa tiene exactamente las mismas instancias que otra; la distancia entre la bolsa A y la bolsa B será simétrica a la distancia entre B y A; y la distancia entre dos bolsas A y B será siempre menor que la suma de dicha distancia y la distancia a una tercera bolsa C. Relajar alguna de estas restricciones hace que se pase de tener una métrica de distancia estricta, a una medida más relajada que también nos permite determinar la disimilitud entre objetos. En este trabajo se discuten varias propuestas para la definición de disimilitudes entre bolsas, mostrando qué suposiciones hacen implícitamente estas definiciones y, por lo tanto, qué definiciones son adecuadas para diferentes tipos de problemas de MIL abordados desde una perspectiva de aprendizaje no supervisado, concretamente *clustering*.

En las siguientes secciones se definen las distintas métricas de disimilitud estudiadas, abordadas desde dos perspectivas atendiendo a la representación de las bolsas de MIL: tratar las bolsas como conjuntos de puntos en un espacio D-dimensional, donde cada punto es una instancia de las que componen la bolsa; o tratar la bolsa como una distribución de las instancias que la componen.

IV-A. *Bolsas como conjuntos de puntos*

El primer enfoque para definir una métrica de disimilitud entre bolsas es considerar cada bolsa como un conjunto de puntos en un espacio altamente dimensional. Partiendo de este enfoque, se pueden definir métricas que utilicen de base una distancia entre instancias clásicas, por ejemplo la distancia euclidiana, y a partir de ésta distancia de base apliquen distintos tratamientos para obtener una distancia a nivel de bolsa.

En esta categoría entra la métrica de distancia más común y utilizada en MIL, la distancia de Hausdorff, H_{max}, que se basa en que dos bolsas están cercanas la una a la otra si cada instancia de una de las bolsas es cercana a una instancia de la otra. Esta cercanía se mide en base a una métrica de distancia d entre instancias, que en este trabajo es la distancia Euclidiana, ampliamente utilizada conjuntamente con Hausdorff [12]. Así, la distancia de Hausdorff utiliza la máxima diferencia entre las instancias de las respectivas bolsas:

$$H_{max}(b_i, b_j) = \max_{x_{ki} \in b_i} \min_{x_{lj} \in b_j} d(x_{ki}, x_{lj}) \quad (2)$$

Los pasos que componen esta distancia aseguran que se cumplen las restricciones de identidad, simetría y desigualdad triangular mencionadas previamente, necesarias para considerar una métrica de disimilitud también como una métrica de distancia.

Sin embargo, H_{max} puede ser muy sensible a datos anómalos [12], [14], [15], por lo que han surgido variaciones que alteran el estadístico del segundo paso una vez se han calculado las distancias mínimas entre las instancias de b_i y b_j: las disimilitudes mínima y media de mínimos que se estudian a continuación. Además, también se estudia una propuesta en la que el primer paso no es considerar únicamente las distancias mínimas, sino que considera todas las distancias en la métrica definida como disimilitud media de Hausdorff. Estas tres variaciones se definen a continuación en esta sección.

La disimilitud mínima de Hausdorff H_{min} se define a continuación como la mínima diferencia entre dos bolsas, utilizando como distancia de base entre instancias la distancia euclidiana:

$$H_{min}(b_i, b_j) = \min_{x_{ki} \in b_i} \min_{x_{lj} \in b_j} d(x_{ki}, x_{lj}) \quad (3)$$

La disimilitud media de mínimos de Hausdorff H_{avmin} se define a continuación como la media de las mínimas diferencias entre dos bolsas, utilizando como distancia de base entre las instancias la distancia euclidiana:

$$H_{avmin}(b_i, b_j) = \frac{1}{|b_i|} \sum_{x_{ki} \in b_i} \min_{x_{lj} \in b_j} d(x_{ki}, x_{lj}) \quad (4)$$

La disimilitud media de Hausdorff H_{mean} se define a continuación como la media de las distancias entre todas las instancias de dos bolsas, utilizando como distancia de base entre las instancias la distancia euclidiana:

$$H_{ave}(b_i, b_j) = \frac{1}{|b_i||b_j|} \sum_{x_{ki} \in b_i} \sum_{x_{lj} \in b_j} d(x_{ki}, x_{lj}) \quad (5)$$

Nótese que tanto H_{min} como H_{avmean} son simétricas pero no tienen por qué cumplir la propiedad de identidad, $H_{min}(b_i, b_j) => b_i = b_j$, si ambas bolsas contienen a la misma instancia. En el caso de H_{avmin} no se cumple la propiedad simétrica, $H_{avmin}(b_i, b_j) \neq H_{avmin}(b_j, b_i)$.

IV-B. *Bolsas como distribuciones de instancias*

Alternativamente, se puede ver cada bolsa como una distribución de probabilidad en el espacio de las instancias, definiendo la disimilitud entre bolsas como una distancia entre sus respectivas distribuciones. Dado que las distribuciones reales no están disponibles, las distribuciones de instancia deben aproximarse.

La primera propuesta en esta línea es aproximar cada bolsa b_i como una distribución normal definida por su vector de medias μ_i y su matriz de covarianzas Σ_i en el espacio de las instancias D. Así, la disimilitud entre dos bolsas se basa en la distancia de Mahalanobis:

$$Maha(b_i, b_j) = (\mu_i - \mu_j)^\top \left(\frac{1}{2}\Sigma_i + \frac{1}{2}\Sigma_j \right)^{-1} (\mu_i - \mu_j) \quad (6)$$

En este caso, se cumple la propiedad de la simetría, pero no tiene por qué cumplirse la de la identidad.

La segunda propuesta se basa en relajar el supuesto de que cada bolsa se aproxima a una distribución normal, asumiendo por contra que se aproxima a una distribución gaussiana multivariante. Así, la disimilitud entre dos bolsas b_i y b_j se basa en el cálculo de la divergencia entre sus distribuciones por medio de la desigualdad de Cauchy–Schwarz:

$$CS(b_i, b_j) = -\log \left(\frac{K_\sigma(b_i, b_j)}{\sqrt{K_{2\sigma}(b_i, b_i) K_{2\sigma}(b_j, b_j)}} \right) \quad (7)$$

Donde K es una función de estimación de densidad kernel dadas dos distribuciones (bolsas) pertenecientes al mismo U con un parámetro σ que determina la anchura de las distribuciones:

$$K_\sigma(b_i, b_j) = \sum_{\substack{x_{ki} \in b_i, \\ x_{lj} \in b_j}} \frac{\exp(\frac{1}{-2\sigma^2}(x_{ki} - x_{lj})^\top (x_{ki} - x_{lj})}{(2\pi\sigma 2)^{D/2}} \quad (8)$$

Por último, reduciendo aún más la anchura de la distribución estimada, cada b_i puede ser representada como un conjunto de funciones delta de Dirac, una en cada una de las instancias que componen la bolsa. Así, para esta representación se propone usar la distancia del movimiento de tierra (*Earth Movers Distance*, EMD), que mide la cantidad mínima de trabajo a aplicar para transformar una distribución de probabilidad b_i, o montón de tierra, en otra distribución de probabilidad b_j, o agujero en el suelo en el que habría que introducir la tierra. Así, si en una bolsa b_i hay m_i instancias, cada instancia tiene $1/m_i$ de la masa total de probabilidad, de forma que la pila se interpreta como la suma de m_i pilas más pequeñas y el agujero como m_i agujeros más pequeños. Con lo que la distancia EMD basada en esta distribución se define como:

$$EMD(b_i, b_j) = \sum_{\substack{x_{ki} \in b_i, \\ x_{lj} \in b_j}} f(x_{ki}, x_{lj}) d(x_{ki}, x_{lj}) \quad (9)$$

Donde $d(x_{ki}, x_{lj})$ es la distancia de base entre agujeros y pilas definida como la distancia euclidiana entre instancias y $f(x_{ki}, x_{lj})$ es el flujo que minimiza la distancia general, que en la práctica es un optimizador lineal algebraico de tipo Simplex y sujeto a las restricciones que aseguren que sólo las cantidades disponibles de tierra son transportadas a sólo los agujeros disponibles, $f(x_{ki}, x_{lj}) \geq 0$, $\sum_{x_{ki} \in b_i} f(x_{ki}, x_{lj} \leq 1/m_j)$ y $\sum_{x_{lj} \in b_j} \leq 1/m_i$; y que se transporta el total de la tierra, $\sum_{x_{ki} \in b_i, x_{lj} \in b_j} f(x_{ki}, x_{lj} = 1)$.

V. Estudio experimental

En esta sección se expone el marco experimental utilizado para el estudio comparativo de los distintos enfoques de métricas de disimilitud entre bolsas en el algoritmo MIKMeans.

V-A. Conjuntos de datos MIL

MIL se ha aplicado satisfactoriamente a numerosas e interesantes aplicaciones relativas a diferentes dominios [3]. En este estudio se considera un amplio rango de conjuntos de datos o *datasets* públicos de MIL, cuyas características se resumen en la Tabla I. Concretamente se tratan de 28 *datasets* pertenecientes a los dominios de predicción del conocido como reto *East-west* sobre determinar el sentido de la marcha de un tren, la predicción de actividad molecular, la anotación de los elementos presentes en imágenes, la identificación de proteínas, la categorización del tema de un texto, la identificación del canto de un pájaro y la evolución temporal de la ocurrencia de fallos de lectura/escritura en discos magnéticos.

Tal y como se estudió en nuestro trabajo previo [13], los valores de los atributos de instancia para cada *dataset* deben ser normalizados con el fin de evitar desviaciones en el cálculo de la distancia entre bolsas cuando las escalas de los atributos difieren. Así, dado el atributo d de la instancia i en la bolsa j, a_{dij}, se ha aplicado la normalización Z_5 [16]:

$$Z_5(y_{dij}) = \frac{a_{dij} - \text{mín}_{a_{dkl} \in U}(a_{dkl})}{\text{máx}_{a_{dkl} \in U}(a_{dkl}) - \text{mín}_{a_{dkl} \in U}(a_{dkl})} \quad (10)$$

V-B. Métricas de validación

Para evaluar y comparar la influencia de cada métrica de disimilitud en la formación de los clústeres se incluyen en este estudio varios de los índices de validación de clustering más representativos que nos permiten analizar la bondad de las soluciones de clustering de acuerdo con la homogeneidad y separación de los clústeres. Además, estas métricas se han adaptado a MIL.

La primera métrica es el índice Silhouette para MIL S_{MI}, que valida el rendimiento de la agrupación basándose en la diferencia por pares de las distancias entre clústeres y dentro de los clústeres. Así, relaciona la distancia media mínima entre una bolsa b_i y todas las bolsas del resto de clusterés y la distancia media entre b_i y todas las demás bolsas de su clúster. Es una métrica a maximizar:

$$S_{MI} = \frac{1}{K} \sum_{k=1}^{K} \frac{1}{|C_k|} \sum_{b_i \in C_k} \frac{Ext_{MI}(b_i) - Int_{MI}(b_i)}{\text{máx}(Ext_{MI}(b_i), Int_{MI}(b_i))} \quad (11)$$

Tabla I
Información de los datasets MIL

Dominio	Dataset	Bolsas		Atribut. de instancia	Media instancias / bolsa
		Pos.	Neg.		
East-West challenge	Eastwest	10	10	24	10.65
	Westeast	10	10	24	10.65
Actividad molecular	Musk1	47	45	166	5.17
	Musk2	39	63	166	64.69
	Suramin	7	4	20	216.18
	Mutag3Atoms	125	63	10	8.61
	Mutag3Bonds	125	63	16	21.25
	Mutag3Chains	125	63	24	28.45
Anotación de imágenes	ImgTiger	100	100	230	6.10
	ImgFox	100	100	230	6.60
	ImgElephant	100	100	230	6.96
	Graz02Bikes	365	380	90	7.00
	Graz02Car	420	380	90	7.00
	Graz02People	311	380	90	7.00
	CorelAfrican	100	1900	9	3.97
	CorelAntique	100	1900	9	3.97
	CorelBattleships	100	1900	9	3.97
	UCSBBreastCancer	26	32	708	34.52
Identificación de proteínas	Thioredoxin	25	168	8	137.88
Categorización de texto	Newsgroups1	50	50	200	54.43
	Newsgroups2	50	50	200	30.94
	Newsgroups3	50	50	200	51.75
	BiocrComponent	423	2707	200	11.79
	BiocrFunction	443	4799	200	10.59
Grabación de audio	BirdsBrownCre	197	351	38	18.67
	BirdsChest	117	431	38	18.67
	BirdsHammo	103	445	38	18.67
Rendimiento en discos magnéticos	Harddrive1	191	178	61	185.40

Donde

$$Ext_{MI}(b_i) = \min_{l, l \neq k} \frac{1}{|C_l|} \sum_{b_j \in C_l} H_d(b_i, b_j) \quad (12)$$

$$Int_{MI}(b_i) = \frac{1}{C_k - 1} \sum_{b_j \in C_k, j \neq i} H_d(b_i, b_j) \quad (13)$$

La segunda métrica estudiada es el índice de Davies-Bouldin para MIL DB_{MI}, que mide simultáneamente tanto la compacidad como la separación de los clústeres formados. Es una métrica a minimizar:

$$DB_{MI} = \frac{1}{K} \sum_{k=1}^{K} \max_{l, l \neq k} \frac{\frac{1}{|C_k|} \sum_{b_i \in C_k} H_d(b_i, c_k) + \frac{1}{|C_l|} \sum_{b_j \in C_l} H_d(b_j, c_l)}{H_d(c_k, c_l)} \quad (14)$$

El tercer índice considerado, S_Dbw, considera la densidad para medir la separación inter-clúster. La idea básica es que para cada par de clústeres C_i y C_j al menos una de sus densidades sea mayor que la densidad del punto medio entre sus centros u_{ij}. Así, el índice se define como la suma de esta separación y la compacidad intra-clúster. Es una métrica a minimizar:

$$S\_Dbw_{MI} = Scat_{MI}(K) + Den_{MI}(K) \quad (15)$$

Donde

$$Scat_{MI}(K) = \frac{1}{K} \sum_{k=1}^{K} \frac{||\sigma(C_k)||}{||\sigma(U)||} \quad (16)$$

$$Den_{MI}(K) = \frac{1}{K^2 - K} \sum_{i=1}^{K} \sum_{\substack{j=1 \\ j \neq i}}^{K} \frac{\sum_{b_k \in C_i \cup C_j} H_d(b_k, u_{i,j})}{\max(\sum_{b_l \in C_i} H_d(b_l, c_i), \sum_{b_m \in C_j} H_d(b_m, c_j))} \quad (17)$$

Por último, se estudia el índice de validación basada en densidad para MIL $DBCV_{MI}$, que se basa en calcular la distancia mutua de accesibilidad (*Mutual Reach Distance*, MRD) entre cada par de bolsas en un clúster y a partir de ahí construye un grafo completamente conectado con las bolsas como nodos y las distancias como pesos de las aristas. Dado este grafo para cada clúster, el siguiente paso es construir un árbol mínimo de expansión, denominado *Minimum Spaning Tree*, MST_{MRD}. Con estos árboles calculados se define el índice a partir de la separación de densidades entre pares de clústeres (el área de máxima densidad entre ambos clústeres) y la escasez de densidad de cada clúster (el área con la densidad más baja del clúster). Es una métrica a maximizar:

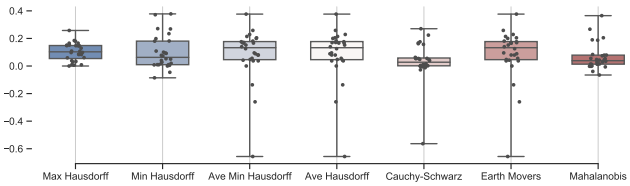
$$DBCV_{MI} = \sum_{k=1}^{K} \frac{|C_i| \min_{j \neq k}(DSp(C_k, C_j)) - DSe(C_k)}{N \max(\min_{j \neq k}(DSp(C_k, C_j)), DSe(C_k))} \quad (18)$$

Donde $DSp(C_k, C_j)$ es la distancia mínima de accesibilidad entre los nodos internos de los MST_{MRD} de ambos clústeres y $DSe(C_k)$ es el máximo peso de las aristas del MST_{MRD} construido para el clúster C_k.

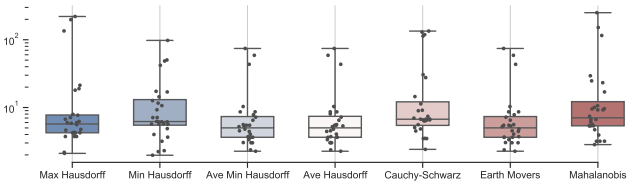
V-C. Resultados

El algoritmo y las medidas de disimilidad se han implementado en el lenguaje de programación Java, haciendo uso de la librería Weka para el tratamiento de individuos MIL y haciendo uso de la librería Apache Commons Math para las funciones estadísticas y matemáticas más avanzadas. En cuanto al único parámetro a especificar del algoritmo MIKMeans, el número de clústeres K, se ha fijado a 2 en todos los casos, que se corresponde con el número de clases que tienen los *datasets* con los que se ha trabajado. Por último, dado que MIKMeans es estocástico, cada experimento se ha repetido con 10 semillas diferentes. En este trabajo se presentan los resultados medios sobre los que finalmente se ha realizado el test estadístico de Friedman que nos permite determinar si estadísticamente existen diferencias significativas en el rendimiento del algoritmo utilizando las diferentes medidas de disimilitud.

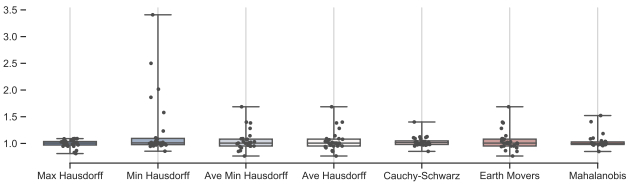
Los resultados de los índices de validación de *clustering* obtenidos por las distintas métricas de disimilitud estudiadas se muestran en la Figura 1 en formato de diagramas de caja para los 28 *datasets* estudiados, permitiéndonos apreciar la dispersión de los resultados obtenidos. En la Tabla II se muestran los rankings obtenidos por el test de Friedman para los distintos índices de validación estudiadas que nos permite visualizar las mejores propuestas, aquellas cuyo valor de ranking sea más cercano a 1. Finalmente, la Tabla III indica los resultados del test de Friedman, incluido el p-value que indica la existencia de diferencias significativas con confianzas superiores al 99 % en los casos de los índices Silhoutte y DB, y superiores al 90 % en los casos de S_Dbw y DBCV. En consecuencia, se ha aplicado también el test *post-hoc* de Shaffer para encontrar los grupos significativamente mejores



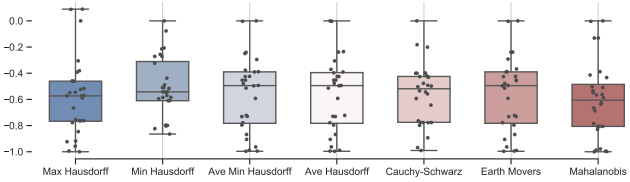
(a) Índice Silhouette para MIL



(b) Índice Davies-Bouldin para MIL



(c) Índice S_Dbw para MIL



(d) Índice DBCV para MIL

Figura 1. Distribución de resultados por métrica de similitud

entre las métricas evaluadas. Los resultados se muestran en la Figura 2.

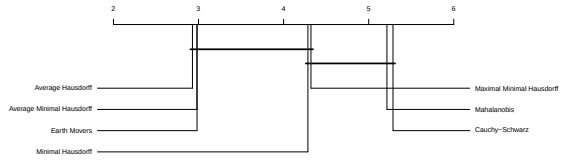
Tabla II
RANKINGS MEDIOS DE LAS DIFERENTES PROPUESTAS

	H_{max}	H_{min}	H_{avmin}	H_{ave}	$Maha$	CS	EMD
S_{MI}	4.29	4.03	3.03	**2.98**	5.39	5.19	3.08
DB_{MI}	4.42	4.29	2.55	**2.60**	5.90	5.65	**2.60**
S_Dbw_{MI}	**3.06**	4.77	3.71	3.76	4.35	4.71	3.63
$DBCV_{MI}$	4.26	3.69	3.55	**3.45**	5.11	4.40	3.53

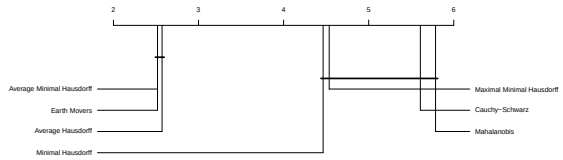
Tabla III
RESULTADOS DEL TEST DE FRIEDMAN

Métrica	S_{MI}	DB_{MI}	S_Dbw_{MI}	$DBCV_{MI}$
p-value	2.302e-07	5.399e-13	0.01833	0.01890
Estadístico	41.506	69.411	15.259	15.180

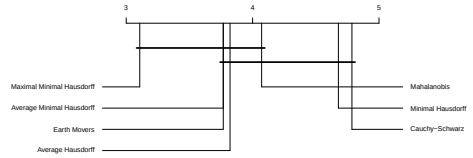
De estos resultados se puede destacar como principal conclusión que, si bien la utilización del enfoque clásico de distancia (H_{max}) puede ser una elección razonable, sobre todo para encontrar clústeres densos (bajo índice S_Dbw),



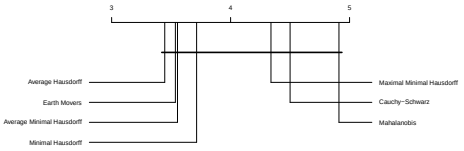
(a) Diferencias al 99 % de conf. del índice S_{MI}



(b) Diferencias al 99 % de conf. del índice DB_{MI}



(c) Diferencias al 90 % de conf. del índice S_Dbw_{MI}



(d) Diferencias al 90 % de conf. del índice $DBCV_{MI}$

Figura 2. Test *post-hoc* de Shaffer

es un enfoque básico que se puede superar utilizando un enfoque basado en disimilitud. Concretamente, se observa una clara ventaja de las métricas que distribuyen la importancia de la disimilitud en todas las instancias que la componen. Así, H_{ave}, que es la única que no evalúa únicamente las distancias mínimas, obtiene los mejores resultados en cuando a forma, compacidad, separación y densidad de los clústeres. Atendiendo al enfoque de bolsas como conjuntos de puntos, a H_{ave} le sigue en rendimiento la siguiente métrica más inclusiva, H_{avmin}, que considera todas las distancias mínimas en conjunto. A un nivel similar de rendimiento se encuentra EMD, que sigue también esta línea de inclusividad, este caso desde la perspectiva de crear una distribución de probabilidad por cada instancia en la bolsa. Las otras métricas de disimilitud basadas en distribuciones obtienen peor rendimiento conforme más resumen la información por bolsa, de forma que en todos los casos $Maha$ se comporta peor que CS al crear una única distribución de probabilidad por bolsa. Por último, la métrica H_{min} que considera únicamente la menor distancia entre dos bolsas tiene un alto grado de variabilidad: en los datasets con más ruido como los relativos a categorización de texto o grabación de audio, son en los que obtiene peores resultados.

VI. Conclusiones y trabajo futuro

Este trabajo realiza un estudio de la influencia de la medida de disimilitud entre bolsas para obtener la mejor agrupación de bolsas en MIL. Concretamente se han estudiado 7 métricas diferentes que nos permiten determinar la disimilitud entre bolsas en MIL con un total de 28 conjuntos de datos. Los resultados determinan que utilizar métricas que tengan en cuenta el mayor número de instancias para computar la disimilitud entre bolsas es positivo desde el punto de vista de la cohesión, densidad y separación de los agrupamientos formados.

Este trabajo abre dos interesantes líneas: aplicar el enfoque de disimilitud entre bolsas en más algoritmos de clustering para MIL y analizar cómo afecta este enfoque de disimilitud en otros índices de validación de clustering relacionadas con atributos extrínsecos al agrupamiento como las clases que las bolsas tienen, cuando estas son conocidas.

Agradecimientos

Este trabajo está financiado por el proyecto de investigación UCO-1262678 de la Junta de Andalucía y el FEDER.

Referencias

[1] D. Xu and Y. Tian, "A Comprehensive Survey of Clustering Algorithms," *Annals of Data Science*, vol. 2, no. 2, pp. 165–193, 2015.

[2] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez, "Solving the multiple instance problem with axis-parallel rectangles," *Artificial Intelligence*, vol. 89, no. 1-2, pp. 31–71, 1997.

[3] F. Herrera, S. Ventura, R. Bello, C. Cornelis, A. Zafra, D. Sánchez-Tarragó, and S. Vluymans, *Multiple Instance Learning*. Springer, 2016.

[4] P. Shamsolmoali, M. Zareapoor, H. Zhou, and J. Yang, "AMIL: Adversarial multi-instance learning for human pose estimation," *ACM Transactions on Multimedia Computing, Communications and Applications*, vol. 16, no. 1s, 2020.

[5] T. Zhang, S. Liu, C. Xu, and H. Lu, "M4L: Maximum margin Multi-instance Multi-cluster Learning for scene modeling," *Pattern Recognition*, vol. 46, no. 10, pp. 2711–2723, 2013.

[6] F. F. Alam and A. Shehu, "Unsupervised multi-instance learning for protein structure determination," *Journal of Bioinformatics and Computational Biology*, vol. 19, no. 1, 2021.

[7] C. He, J. Shao, J. Zhang, and X. Zhou, "Clustering-based multiple instance learning with multi-view feature," *Expert Systems with Applications*, vol. 162, 2020.

[8] C. Henegar, K. Clément, and J. D. Zucker, "Unsupervised multiple-instance learning for functional profiling of genomic data," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 4212 LNAI, 2006, pp. 186–197.

[9] H.-p. Kriegel, A. Pryakhin, and M. Schubert, "An EM-Approach for Clustering Multi-Instance Objects," in *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, 2006, pp. 139–148.

[10] H. P. Kriegel, A. Pryakhin, M. Schubert, and A. Zimek, "COSMIC: Conceptually specified multi-instance clusters," in *Proceedings - IEEE International Conference on Data Mining, ICDM*, 2006, pp. 917–921.

[11] M. L. Zhang and Z. H. Zhou, "Multi-instance clustering with applications to multi-instance prediction," *Applied Intelligence*, vol. 31, no. 1, pp. 47–68, 2009.

[12] V. Cheplygina, D. M. Tax, and M. Loog, "Multiple instance learning with bag dissimilarities," *Pattern Recognition*, vol. 48, no. 1, pp. 264–275, 2015.

[13] A. Esteban, A. Zafra, and S. Ventura, "A Preliminary Study on Evolutionary Clustering for Multiple Instance Learning," in *Proceedings of IEEE Congress on Evolutionary Computation*, Glasgow, UK, 2020, pp. 1–8.

[14] M.-P. Dubuisson and A. Jain, "A modified Hausdorff distance for object matching," in *International Conference on Pattern Recognition*, 2002, pp. 566–568.

[15] C. Zhao, W. Shi, and Y. Deng, "A new Hausdorff distance for image matching," *Pattern Recognition Letters*, vol. 26, no. 5, pp. 581–586, 2005.

[16] M. C. Cooper and G. W. Milligan, "A study of standardization of variables in cluster analysis," *Journal of Classification*, vol. 5, no. 2, pp. 181–204, 1988.

A cluster merge algorithm to classify on/offline records of researchers in programming training

Sonia Estévez
Complutense University
Computer Science Faculty
28040 Madrid
Email: soesteve@ucm.es

Victoria López
CUNEF University
Quantitative Methods Dep.
28040 Madrid
Email: victoria.lopez@cunef.edu

***Abstract*—In recent years, the online work of researchers has developed enormously. An interest in programming and data analysis has also been observed in researchers from almost all areas of knowledge. However, the diversity of profiles constitutes an impediment to finding generalized training solutions. This article presents a study of a dataset of 180 researchers (96 women and 84 men) from the Complutense University of Madrid. The participants were monitored during an online training period. The analysis of the homogeneity of the data and an algorithm to balance the non-successful results of traditional clustering algorithms are presented. The homogeneity of the data regarding the gender of the participants as well as knowledge areas have been studied. Regarding the technological profile of the participants, two clustering models have been proposed: K-Means and hierarchical clustering. However, none of them is adequate to our interest due to unbalance number of clusters or unbalance cardinality. Therefore, a merge algorithm is presented to balance the clusters. This algorithm is based on the output of the two mentioned algorithms. The algorithm is applied to a reduced dataset to illustrate its behaviour.The result is a more appropriate classification for the creation of workgroups: the number of clusters is smaller and the cluster sizes are more balanced.**

I. Introduction

The research work demands a big knowledge of data science. Not only researchers in the areas of engineering, physics and mathematics but also almost all areas (social, health, biology, etc.) are very interested in learning programming languages, that allow them to analyze the data from their experiments [1], [2].

Research projects require interdisciplinary groups with common interests. For this reason, training suited to the interests of the team is more effective than general training.

The objective of this work is the creation of homogeneous (regarding the profile of the participants) and balanced workgroups in size. Classical classification methods have been used. However, although the results are robust, they are not satisfactory since many groups of small size or few groups of very different sizes are obtained. To solve the problem, some proposals in the literature have been analyzed as described below.

A. Stay of Art

The article [3] proposes a combined method with hierarchical clustering based on dendrograms. Dendrograms are described by matrices that are operated to obtain a derived classification. Also, in [4], the authors proposed an improvement classification by combining the largest minimum distance algorithm and the K-Means algorithm to propose a K-Means clustering algorithm that avoids cluster atomization. In the same line, the article [5] proposes to combine multiple groupings to improve the robustness and stability of the results. To do this, it proposes an iterative voting process. In the present work, a new alternative is presented to merge clustering by combining dendrogram results with K-Means results. The Merge-Cluster algorithm has been tested on the data set described before.

The structure of the paper is as follows. Section 2 details the data collected and its structure. Section 3 explains the preliminary analysis including the homogeneity of the data concerning the gender of the participants. Section 4 shows the classification methods applied and the merge-cluster algorithm. Section 5 contains the results obtained over a reduced set of data. The paper ends with conclusions and future work.

II. Data sources and structure

This work has been carried out with data from researchers enrolled in R programming [6], [7] and data analysis courses for employees of the Complutense University of Madrid (PDI training). The data collected has been completely anonymized and only reflects the area of knowledge, gender and the use of on/offline computer resources.

Three datasets have been collected: the first one is monitored data from a virtual campus (Moodle 3.8 [8]), with a total of 26590 observations of 10 variables. The second one is a dataset provided by the university with some personal data of the participants as faculty or professional category. The third data set was collected by satisfaction surveys (with Google forms created by the authors of the present work) and it provides subjective information as usability and valuation of the training. The data collected has been previously processed and integrated into a single data set free of personal data. This data set is the one used for this study, therefore, no special authorization is required for the publication of the results. In addition, the participating researchers have been informed of this study and have been able to exercise their right to delete their data.

Table I shows the characteristics of this training during the years 2018-2020. In total, 9 editions have been carried out, 3 completely face-to-face and 6 completely online, the latter with fewer hours of training (10 less each). This table also shows the specific number of men and women participating in each edition.

Although the training is very specific, the profiles of the researchers are very varied. Table II shows the count of researchers by faculties. 21 faculties represent the different areas of knowledge. Women are present in 19 of them and men in 17.

III. Preliminar Analysis

Two studies are conducted in this section. The first is a priori analysis. This analysis is done with the registration data of participants. Researchers' interest in data science training is studied and answers the question of whether or not there is a difference between genders. The second is a posteriori analysis. This analysis is carried out with the data collected automatically in the system during the training period and studies the differences in behaviour between men and women. All the calculations and algorithms of the following sections have been developed in R with the help of RStudio [7].

A. Significance of gender a priori

To analyze the significance of the gender, Student's t-test [9] has been applied to analyze the gender difference in the following five cases: (a) all participants of all courses; (b) participants of online courses; (c) participants of offline courses (face-to-face); (d) participants by faculty; (e) participants by professional category. The data is shown in Table III. Although the data reveal that there are no significant differences between the number of participants of each sex (p-values greater than $\alpha = 0.05$), we can affirm that the presence of women tend to be greater in all the cases: the confidence intervals obtained have more weight on the negative side than on the positive side and the means of the differences (last column of the table) are also all negative. However, from a statistical point of view, there are no significant differences. Thus, the results of this preliminary study show that the sample is homogeneous with respect to gender.

B. Significance of gender a posteriori

Gender differences in the use of technologies have been studied here. The most significant variables are analyzed during training: access to resources, the performance of tasks, system logins and online collaboration. This last variable is only analyzed in the case of online training since it is not significant in face-to-face training.

Figure 1 shows the boxplot diagrams [10] of the first three variables: Tasks, Resources and System. Each variable is studied for face-to-face training and online training. The values of the medians marked with a line inside the boxes are an indicator that the data are more homogeneous when they come from online training, and this occurs in both genders (although the change is higher in men). In all cases, an increase in the use of technologies is observed when training is online. The differences between genders remain: Women use more resources than men and have more entries into the system. In terms of tasks, both have similar use. Regarding the assessment of the training received, 80% of the women surveyed give the highest score, while in men it is the only 60%. Regarding the preference of the type of training (face-to-face, mixed or online), women are more open to mixed or online training than men. 65% of women prefer mixed training, while 68.8% of men prefer face to face training. Totally online training is only chosen by 17% of women and 2.8% of men.

Therefore, we can conclude that men and women present very different behaviour regarding the use of the technologies available for online training. In the next section, more variables are added to the study.

IV. Classification Algorithms

Classification methods are especially useful for detecting interest groups not visible through previously established categories. In this section, the researchers are classified taking into account 7 behavioural variables in the use of technologies. These variables contain the number of accesses of each user to the different technological resources: Sending files, creating folders, accessing resources, accessing the system, performing tasks, accessing URLs and accessing Collaborate.

A classification into k groups is the assignment of a value from 1 to k to each of the observations in a dataset. The algorithms use the similarity of the variables to assign the group corresponding to each observation. Observations with similar values fall into the same group and observations with very different values must fall into different groups. In this work, we have used two traditional classification methods: Single Linkage (and dendrograms) and K-Means. Both of them are based on distances and they are briefly explained as follows.

The Single Linkage algorithm [11] defines the distance L between two clusters A, B as the smallest pairwise distance among elements from each cluster, that is:

$$L(A, B) = min_{x \in A, y \in B} dist(x, y)$$

A dendrogram is a binary tree with a set of observations at its leaves. Whatever internal node represents a subset (or cluster) of the leaves that hold as children of it. The level of each node is the distance from the root to the node. The hierarchical agglomerative clustering is performed in R by "hclust" function. The input is the set of observations and the output is the descriptive dendrogram representing the clustering. Starting with one singleton cluster per observation, each step, find the clusters X and Y with the lowest linkage L. They merge by creating a parent at a new level L. The process ends when all the observations are in one cluster.

K-Means algorithm [4], [11] is a heuristic one that tries to solve the NP-Complete problem of the K-Means. Starting with a random of K selected observations, and the K corresponding clusters of the nearest observations. The algorithm iterates partitioning the data using the nearest centroid decision

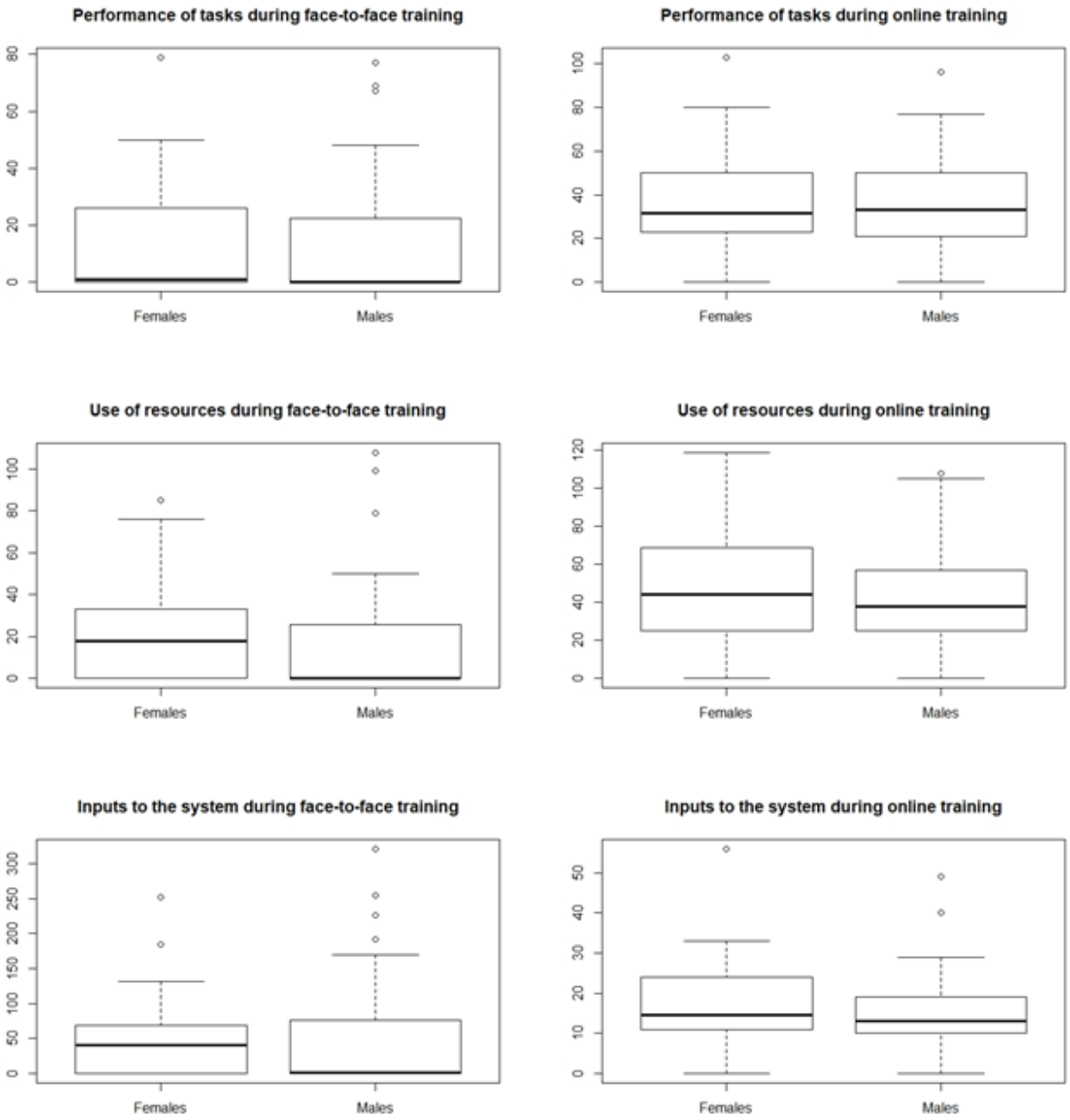


Fig. 1. Boxplots showing the variability of the data per gender and on/offline training

TABLE I
TRAINING CHARACTERISTICS

Edition	Training type	Modality	Participants	Hours	Female	Male
2018/Sept	Learning R and RStudio	Offline	20	25	13	7
2019/jun	Learning R and RStudio	Offline	21	25	9	12
2020/jan	Learning R and RStudio	Offline	20	25	12	8
2020/jun	Learning R and RStudio	Online	19	15	12	7
2020/jun	Data transformation	Online	20	15	10	10
2020/jun	Algorithms & Machine Learning	Online	20	15	6	14
2020/sept	Learning R and RStudio	Online	20	15	13	7
2020/sept	Data transformation	Online	20	15	13	7
2020/sept	Algorithms & Machine Learning	Online	20	15	8	12
			180		96	84

TABLE II
PARTICIPANTS FOR FACULTY AND GENDER

Faculty	Female	Male
Arts	1	1
Biology	10	5
Business	12	15
Chemistry	1	3
Marketing	1	0
Documentation	1	0
Education	9	1
Geography	4	4
Geology	4	7
Informatics	0	1
Journalism	4	3
Laws	0	1
Math	1	3
Medicine	5	11
Others	1	1
Pharmacy	7	2
Philology	3	0
Physics	7	6
Politics	2	3
Psycholgy	9	10
Statistics	1	2
Veterinary	13	5
	96	84

rule. Each step, it recalculates the centroids of each cluster until no changes. The objective is to minimize the variance within groups (measured with the variable $tot.withinns$) and maximize the variance between groups (measured with the variable $betweens$). The best classification is reached with the K value that maximizes the accuracy given by the quotient $betwees/tot.withinns$. The disadvantage of this method is that the value of K must be known beforehand.

A. Merge-Cluster Algorithm

This section describes the class merge process. The process starts with the partitions outputted by K-Means and Single Linkage algorithms. The output of the K-Means algorithm is denoted as a partition of size k of the observations set, that is

$$C = \{C_1, \ldots, C_k\}$$

A dendrogram clustering of level α can be represented as a set of clusters D_α which is a partition of the set of observations as follows.

$$D_{\alpha-cut} = \{D_1, \ldots, D_M\}$$

Each observation can be represented as a word on the alphabet $A = \{r, l\}$. A^* is denoted to the set of all the words on the alphabet A. An observation corresponds to a sequence of letters indicating their position in the dendrogram: r represents the right child and l represents the left child in the binary tree. For example, the observation $o = rrrllr$ is the observation located in the tree in the $right{-}right{-}right{-}left{-}left{-}right$ position descending from the root to the leaves of the dendrogram.

Two observations o_1, o_2 belong to the same cluster D_i if and only if they have a common prefix of size $\alpha = 1 + \#connections(\alpha - cut)$, that is,

$$o_1, o_2 \in D_i <=> \exists u, v, w \in A^*, o_1 = u.v, o_2 = u.w, |u| = \alpha$$

Let C and D_α the K-Means and Single Linkage clustering respectively. The Merge-Clustering Algorithm is as follows.

- Step 0. Set an initial set $MR := C$ and $p := 1$. Input N ≤ 1.
- Step 1. $\forall i \in \{1, \ldots, k\} \forall j \in \{i \ldots k\}$: if $\exists o_1 \in C_i, \exists o_2 \in C_j : |o_1| = |o_2| = n$ or $|o_1| = |o_2| - 1 = n$ and $o_1[1 : n - p] = o_2[1 : n - p]$ (they have a common prefix of length n-p+1), then merge

  $$C_{ij} = C_i \cup C_j$$

  and update

  $$MR = MR - \{C_i, C_j\} \cup \{C_{ij}\}$$

- Step 2. Set $p := p + 1$
- Step 3. if $p < N$ go to Step 1 otherwise go to Step 4.
- Step 4. Merge into a single new cluster all clusters that remain unchanged and have size smaller than desired. That is,

  $$MR_{K2} = \bigcup_{CL \in C \cap MR} CL$$

  where $K2 = K - |MR - C| + 1$ and stop. The output remains:

  $$MR = \{MR_1, ..., MR_{K2}\}$$

V. RESULTS

To illustrate the results better, a reduced set of observation is filtered: the subset of female online training. The data of the interquartile range relative to the number of connections have

TABLE III
STUDY OF DIFFERENCES BY GENDER

Model	Confidence Interval ($\alpha = 0.05$)	p-value	df*	mean**
By training (all)	(-5.382544, 2.715877)	0.4694	8	-1.333333
By off-line training	(-14.072910, 9.406243)	0.4825	3	-2.333333
By on-line training	(-7.003325, 5.336658)	0.7426	5	-0.8333333
By Faculty	(-2.0796880 , 0.9887789)	0.4679	21	-0.5454
By Professional Category	(-14.663, 10.663)	0.7015	5	-2

* degree of freedom
** mean of the difference (#men - #women)

been chosen to facilitate the visualization. The set contains a total of 30 observations.

Figure 2 shows the dendrogram obtained. The figure shows four groups at cut $\alpha = 35$, with 6, 17, 1, 6 elements respectively. The dendrogram obtained shows few groups, very numerous and with a lot of transversality in the areas of knowledge.

Before showing the results of the K-Means algorithm, it has been verified that the most suitable number of groups for this algorithm is k = 12 as is explained as follows. Figure 3 shows the precision achieved by the algorithm for values of k between 4 and 14. The calculated value is the quotient between the variable $betweenss$ and the variable $tot.withinss$. The first explains the difference between elements in different groups (must be maximized) and the second explains the difference between elements in the same group (must be minimized). The quotient, therefore, is a variable to be maximized. The graph in Figure 3 shows how this variable increases until k = 12 but then decreases, which means that 12 is the optimal number of clusters.

The results of the K-Means algorithm for k = 12 using the notation proposed in the previous section are the family itself.
$C = \{C_1, \dots\ C_{12}\}$
and every cluster:
$C_1 = \{O_1, O_{18}\}, C_2 = \{O_2, O_4\},$
$C_3 = \{O_3, O_6, O_{17}, O_{19}\}, C_4 = \{O_5, O_{12}\},$
$C_5 = \{O_7, O_{23}, O_{25}\}, C_6 = \{O_8, O_9, O_{13}, O_{14}\},$
$C_7 = \{O_{10}, O_{21}, O_{26}, O_{29}, O_{30}\}, C_8 = \{O_{11}, O_{24}\},$
$C_9 = \{O_{15}, O_{20}, O_{22}\}, C_{10} = \{O_{16}\},$
$C_{11} = \{O_{27}\}, C_{12} = \{O_{28}\}$

The results of the Single Linkage algorithm using the notation proposed in the previous section returns the set:
$D = \{O_1 = "llrll", O_2 = "llll", O_3 = "lrrrrrlrl",$
$O_4 = "lllr", O_5 = "llrrl", O_6 = "lrrrrrlrr",$
$O_7 = "rrlrl", O_8 = "lrrrrrrll", O_9 = "lrrrrrrlr",$
$O_{10} = "lrrrlrr", O_{11} = "lrrll", O_{12} = "llrrr",$
$O_{13} = "lrrrrrrrrl", O_{14} = "lrrrrrrrrr",$
$O_{15} = "rrrrl", O_{16} = "rl", O_{17} = "lrrrrl", O_{18} = "llrlr",$
$O_{19} = "lrrrrrll", O_{20} = "rrrl", O_{21} = "lrrrlll",$
$O_{22} = "rrrrr", O_{23} = "rrll", O_{24} = "lrrlr",$
$O_{25} = "rrlrr", O_{26} = "lrrrlrl", O_{27} = "lrll",$
$O_{28} = "lrlr", O_{29} = "lrrrllr", O_{30} = "lrrrrrrrl"\}$
where for $\alpha = 32$ (see 2 returns the following clusters:
$D_1 = \{O_1 = "llrll", O_2 = "llll", O_4 = "lllr",$
$O_5 = "llrrl", O_{12} = "llrrr", O_{18} = "llrlr"\}$
(with ll as common prefix, 6 observations).
$D_2 = \{O_3 = "lrrrrrlrl", O_6 = "lrrrrrlrr",$
$O_8 = "lrrrrrrll", O_9 = "lrrrrrrlr",$
$O_{10} = "lrrrlrr", O_{11} = "lrrll",$
$O_{13} = "lrrrrrrrrl", O_{14} = "lrrrrrrrrr",,$
$O_{17} = "lrrrrl", O_{19} = "lrrrrrll", O_{21} = "lrrrlll",$
$O_{24} = "lrrlr", O_{26} = "lrrrlrl", O_{27} = "lrll",$
$O_{28} = "lrlr", O_{29} = "lrrrllr", O_{30} = "lrrrrrrrl"\}$
(with lr as common prefix, 17 observations).
$D_3 = \{O_{16} = "rl"\}$
(with rl as common prefix, one observation)
$D_4 = \{O_7 = "rrlrl", O_{15} = "rrrrl", O_{20} = "rrrl",$
$O_{22} = "rrrrr", O_{23} = "rrll", O_{25} = "rrlrr"\}$
(with rr as common prefix, 6 observations).

The result of applying the Cluster-Merger Algorithm with $N = 2$ loops, retuns a new partition of 4 clusters:
$CM = \{CM_1, CM_2, CM_3, CM_4\}$
where
$CM_1 = \{O_1, O_2, O_4, O_5, O_{12}, O_{18}\}$
$CM_2 = \{O_7, O_{15}, O_{20}, O_{22}, O_{23}, O_{25}, O_{28}\}$
$CM_3 = \{O_3, O_6, O_8, O_9, O_{10}, O_{13}, O_{14}, O_{17}, O_{19}, O_{21},$
$O_{26}, O_{29}, O_{30}\}$
$CM_4 = \{O_{11}, O_{16}, O_{24}, O_{27}\}$
The cardinality of these clusters is 6, 7, 13, 4 respectively, which is a better distribution than the 12 clusters of the K-Means algorithm or the 4 unbalance clusters of the Single Linkage algorithm.

VI. CONCLUSION

In this work, a solution is presented to classify researchers according to the characteristics collected during online training. The purpose is to establish groups with homogeneous profiles to improve training results. The data is homogeneous for gender and other variables such as the researchers' area of knowledge and the professional category. Although the data set is adequate to apply classification regarding the differences that the subjects present in technological aspects, the traditional classification methods do not satisfy the expected requirements. To improve the results, a class fusion algorithm is proposed.

To better illustrate the results, only the filtered data of women with the variable "number of connections to the system" within the interquartile range are used. The K-Means algorithm returns a partition with too many small classes and the hierarchical clustering algorithm returns a partition

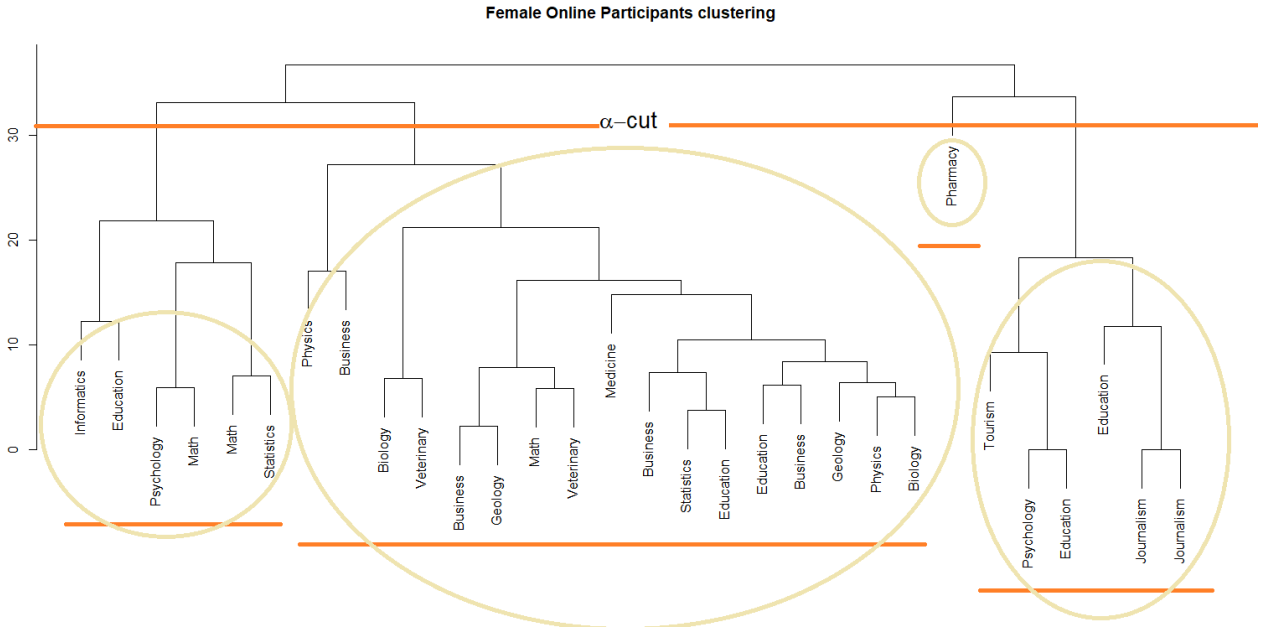


Fig. 2. Female Online Participants herarchical clustering

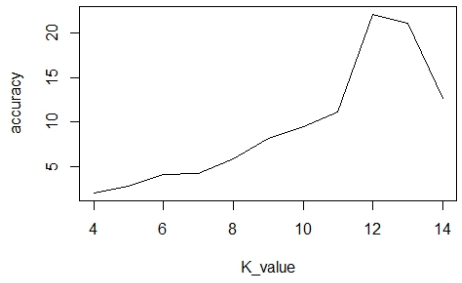


Fig. 3. Accuracy according to the number of clusters

with fewer classes but does not achieve cardinal balance. A merging algorithm is proposed to improve the two classifications (partition 1 and partition 2 respectively). The algorithm uses the information from partition 2 to decide how to merge certain classes from partition 1. The algorithm begins with two classifications, the first resulting from the application of the K-Means algorithm with 12 centers (and small cardinal classes) and the second a result of the application of the hierarchical clustering algorithm with 4 classes with 1, 6, 6, 17 elements (disparate cardinal classes). The final result returns four classes of 6, 7, 13, and 4 elements each, and therefore a more balanced number of elements.

Acknowledgment

This research has been supported by the project "RISE Women with disabilities In Social Engagement (RISEWISE)" under the Horizon 2020 programme (Grant Agreement: 690874) and also by the Spanish MINECO/FEDER project FAME (RTI2018-093608-B-C31) and the Comunidad de Madrid project FORTE-CM (S2018/TCS-4314) co-funded by EIE Funds of the European Union.

References

[1] R. Garcia-Milian, D. Hersey, M. Vukmirovic, and F. Duprilot, "Data challenges of biomedical researchers in the age of omics," *PeerJ*, vol. 6, p. e5553, 2018.

[2] C. P. Chen and C.-Y. Zhang, "Data-intensive applications, challenges, techniques and technologies: A survey on big data," *Information sciences*, vol. 275, pp. 314–347, 2014.

[3] A. Mirzaei, M. Rahmati, and M. Ahmadi, "A new method for hierarchical clustering combination," *Intelligent data analysis*, vol. 12, no. 6, pp. 549–571, 2008.

[4] Y. Li and H. Wu, "A clustering method based on k-means algorithm," *Physics Procedia*, vol. 25, pp. 1104–1109, 2012, international Conference on Solid State Devices and Materials Science, April 1-2, 2012, Macao. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1875389212006220

[5] S. Khedairia and M. T. Khadir, "A multiple clustering combination approach based on iterative voting process," *Journal of King Saud University-Computer and Information Sciences*, 2019.

[6] R Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2021. [Online]. Available: https://www.R-project.org/

[7] RStudio Team, *RStudio: Integrated Development Environment for R*, RStudio, Inc., Boston, MA, 2021. [Online]. Available: http://www.rstudio.com/

[8] "Moodle 3.8," 2021. [Online]. Available: https://www.ucm.es/campusvirtual/formacion-sobre-moodle

[9] W. H. Press, H. William, S. A. Teukolsky, W. T. Vetterling, A. Saul, and B. P. Flannery, *Numerical recipes 3rd edition: The art of scientific computing*. Cambridge university press, 2007.

[10] T. A. Williams, *Statistics for business and economics*. Cengage Learning, 2020.

[11] P. Flach, *Machine Learning: The Art and Science of Algorithms that Make Sense of Data*. Cambridge University Press, 2012.

Valoración de Anomalías Multi-paso Basada en Histogramas para Detección de Anomalías no Supervisada

1st Ignacio Aguilera-Martos
DECSAI[1], DaSCI[2]
Universidad de Granada
Granada, España
nacheteam@ugr.es

2nd Marta García-Barzana
ArcelorMittal Global R&D, New Frontier Digital Portfolio
marta.garcia-barzana@arcelormittal.com

3rd Diego García-Gil
DECSAI[1], DaSCI[2]
Universidad de Granada
Granada, España
djgarcia@decsai.ugr.es

4th Jacinto Carrasco
DECSAI[1], DaSCI[2]
Universidad de Granada
Granada, España
jacintocc@decsai.ugr.es

5th David López
DECSAI[1], DaSCI[2]
Universidad de Granada
Granada, España
derwey@correo.ugr.es

6th Manuel Arias-Rodil
ArcelorMittal Global R&D, New Frontier Digital Portfolio
manuel.arias@arcelormittal.com

7th Julián Luengo
DECSAI[1], DaSCI[2]
Universidad de Granada
Granada, España
julianlm@decsai.ugr.es

8th Francisco Herrera
DECSAI[1], DaSCI[2]
Universidad de Granada
Granada, España
herrera@decsai.ugr.es

***Resumen*—La detección de anomalías es la tarea de detectar muestras que se comportan de forma diferente al resto de los datos o que incluyen valores anormales. La detección de anomalías no supervisada es el escenario más extendido, lo que significa que los algoritmos no pueden entrenar con una entrada etiquetada y no conocen el comportamiento anómalo de antemano. Los métodos basados en histogramas son uno de los enfoques más populares y ampliamente utilizados, destacando con un buen rendimiento y un bajo tiempo de ejecución. A pesar del buen rendimiento, no son capaces de procesar flujos de datos mientras actualizan su conocimiento y manejar una gran cantidad de muestras. En este trabajo proponemos un nuevo enfoque basado en histogramas para abordar los problemas mencionados, introduciendo la capacidad de actualizar la información dentro de un histograma. Hemos aplicado estas estrategias para diseñar un nuevo algoritmo llamado Valoración de Anomalías Multi-paso Basada en Histogramas (VAMBH), que incluye cinco nuevos mecanismos de actualización de histogramas. VAMBH ha sido validado utilizando la biblioteca ODDS como caso general de uso. Se ha utilizado un problema de ingeniería real proporcionado por la empresa multinacional ArcelorMittal para validar el rendimiento en un escenario real. Los resultados han mostrado la validez de VAMBH así como de las estrategias propuestas en términos de rendimiento y tiempos de cómputo.**

***Index Terms*—Histograma, Detección de anomalías, Aprendizaje no supervisado, Series temporales**

I. Introducción

En conjuntos de datos reales encontramos instancias que no siguen el comportamiento o patrón esperado. Estas son conocidas como anomalías, usualmente definidas como datos que no se comportan como la mayoría de las muestras, o los valores que contienen no son normales o comunes [3], [4]. La detección de anomalías afronta diversas situaciones como la salud [10], la vigilancia de redes [11], [13], la monitorización de sensores IoT [14] o las anomalías industriales [8].

Las aplicaciones de maquinaria y monitorización proporcionan los datos finales en formato de serie temporal, requiriendo algoritmos capaces de ingerir este tipo de información [12]. Estas aplicaciones suelen arrojar una gran carga de muestras, lo que requiere una respuesta rápida por parte de los algoritmos. Esta situación ha creado la necesidad de contar con algoritmos rápidos y eficientes, capaces de lidiar con una gran cantidad de muestras y actualizar sus conocimientos mientras reciben nuevos datos.

Las características de la detección no supervisada de anomalías han popularizado herramientas como los histogramas, utilizados en algoritmos como Light Online Detector of Anomalies (LODA) [5] o Histogram Based Outlier Scores (HBOS) [1]. Los detectores de anomalías no supervisados basados en histogramas (como HBOS) suelen tener como inconvenientes: incapacidad de actualizar su estado interno o conocimiento y falta de mecanismos para procesar grandes cantidades de muestras de forma eficiente.

Proponemos un nuevo algoritmo que extiende a HBOS y aborda los inconvenientes mencionados, denominado Valoración de Anomalías Multi-paso Basada en Histogramas (VAMBH). VAMBH está orientado a la actualización rápida y eficiente de la información dentro de un histograma. Incorpora tres mecanismos de actualización para los histogramas convencionales y dos para los histogramas dinámicos introducidos por los autores de HBOS [1]. Los cinco mecanismos nuevos, que se analizan en profundidad en la Sección III, son: estrategia min-max para conjuntos de datos en los que se conoce el mínimo y el máximo absolutos, adición de casillas para mantener la anchura de los intervalos, actualización de la frecuencia ponderada para preservar el número original de

[1]DECSAI Department of Computer Science and Artificial Intelligence
[2]DaSCI Andalusian Institute of Data Science and Computational Intelligence

intervalos, límites dinámicos para preservar el porcentaje de anchura de los intervalos originales sobre el dominio y fusión dinámica para preservar las densidades de los intervalos.

Para validar la calidad y el rendimiento de VAMBH lo compararemos con el algoritmo base HBOS. Estos algoritmos se compararán utilizando la biblioteca Outlier Detection Datasets (ODDS) [2] y un conjunto de datos de ingeniería real proporcionado por ArcelorMittal. VAMBH ha obtenido buenos resultados, mostrando que la propuesta es capaz de detectar con éxito y eficiencia las anomalías en términos de precisión, exhaustividad, F1 y ROC-AUC.

En la Sección II se describe en detalle el problema de detección de anomalías. La Sección III propone VAMBH y todos los mecanismos de actualización del histograma. En la Sección IV se expone el marco experimental utilizado para, a continuación, analizar el rendimiento de VAMBH. Por último, en la Sección V se exponen las conclusiones de este trabajo.

II. Problema de Detección de Anomalías

En esta sección revisaremos el problema de la detección de anomalías describiendo los diferentes tipos de anomalías y sus escenarios de aplicación (Sección II-A). En la Sección II-B, se presentará el estado del arte en la detección probabilística de anomalías no supervisada. En la Sección II-C, se explica en profundidad el algoritmo HBOS.

II-A. Detección de anomalías

Una anomalía es una muestra que no sigue el patrón principal de los datos. Este problema puede ser causado por dos escenarios diferentes [4]. Las muestras anómalas tienen valores extremos, estando alejadas del resto de los datos o bien las muestras anómalas no se alejan espacialmente del resto, pero contienen valores que no son comunes o no siguen el patrón de los datos circundantes. Lo más común es tener anomalías espaciales en conjuntos estáticos y patrones anormales en series temporales.

Hay tres tipos diferentes de problemas cuando se trata de la detección de anomalías [3]. La detección de anomalías *supervisada* en la que los datos de entrenamiento están etiquetados y nuestro algoritmo puede aprender una frontera de separación. La detección de anomalías *semisupervisada* en la que nuestros datos de entrenamiento tienen etiquetadas las muestras normales o casi todos los datos de entrenamiento son normales. Y finalmente la detección de anomalías *no supervisada* en la que las técnicas no tienen una entrada etiquetada.

La salida de un detector de anomalías puede ser: puntuaciones para cada una de las muestras con las que se cuantifica su desviación de los datos normales o bien etiquetas binarias para cada una de las muestras de los datos de entrada. Una forma habitual de obtener estas etiquetas es umbralizar las puntuaciones en función de las estadísticas y de la cantidad aproximada de anomalías que tiene nuestro conjunto de datos. Algunos métodos de umbralización notables son el umbral fijo para datos estáticos y el umbral de ventana deslizante para datos de series temporales.

II-B. Detección de anomalías probabilística no supervisada

La mayoría de los conjuntos de datos y problemas de detección de anomalías entran en la categoría de detección de anomalías no supervisada, ya que no es fácil etiquetar las anomalías. Los algoritmos no supervisados son más aplicables y, por tanto, más utilizados en problemas reales.

De todas las propuestas para detección de anomalías no supervisada, podemos destacar los algoritmos probabilísticos [4]. Entre ellos uno de los más populares y usado es Histogram-Based Outlier Scores (HBOS) [1]. HBOS obtiene un histograma para cada característica que modela la distribución subyacente de ese atributo. Cuando se evalúa una muestra se comprueba la probabilidad de aparición de cada valor en cada histograma. Estas probabilidades se suman, se invierten y se combinan mediante la función logaritmo. Una menor probabilidad implica mayor puntuación y viceversa.

II-C. Histogram Based Outlier Scores

Los histogramas se utilizan para describir la distribución subyacente desconocida de las variables experimentales, dividiendo el espacio en intervalos y contando la cantidad de instancias que tienen valor en ese rango. Podemos expresar el recuento o la frecuencia de forma probabilística escalando el histograma por el valor máximo o la suma de todas las frecuencias absolutas.

El algoritmo HBOS [1] utiliza estos histogramas para extraer información de las características del conjunto de datos. El algoritmo genera un histograma por característica, de modo que podemos obtener una descripción sencilla del mismo. Al final del procedimiento terminamos con tantos histogramas como características tenga el conjunto de datos.

Para obtener una puntuación debemos mezclar esta información. Para ello, los autores de HBOS eligieron la función logaritmo en base dos. En esta fase, los histogramas de cada característica ya están construidos y escalados dividiendo por el recuento más alto, alcanzando su máximo valor en uno.

Para evaluar una instancia p de un conjunto de datos X^d siendo d el número de características, la puntuación de anomalía viene dada por la expresión $HBOS(p) = \sum_{i=1}^{d} log(\frac{1}{histogram(p_i)})$ donde p_i es la i-ésima característica de la muestra p.

Histogramas Dinámicos

Los autores de HBOS propusieron otra alternativa para calcular el histograma, que llamaron “Histograma Dinámico”. Esta modificación surge para mitigar algunos problemas que tienen los histogramas normales.

En el histograma tradicional tomamos un número fijo de intervalos de igual anchura para dividir el espacio y empezar a contar los valores, siendo posible que alguna casilla se quede vacía. En su propuesta, la rutina del histograma intenta mantener todas las casillas con la misma altura, mientras que el ancho será el valor que cambie. Los datos utilizados para construir el histograma se ordenan y se dividen en trozos de igual tamaño. Estos trozos serán las casillas, por lo que tendrán la misma o aproximadamente la misma frecuencia y diferentes

anchos. Si se encuentra un valor repetido la rutina lo coloca en la misma casilla para tener intervalos bien definidos.

Con esta variación, la atención se centra en la anchura de la casilla y no en la frecuencia o la altura. Si la anchura de una casilla es menor, significa que concentra una mayor densidad de valores y, por tanto, éstos son más comunes. Si la anchura de la casilla es mayor, la densidad es menor y, por tanto, la probabilidad será menor.

III. Multi-step Histogram Based Outlier Scores

Esta sección está dedicada a describir completamente el algoritmo VAMBH. Los mecanismos de actualización de histogramas se explicarán en la Sección III-A. Finalmente, el algoritmo propuesto que incorpora dichos mecanismos (VAMBH) se explica en la Sección III-B.

III-A. Mecanismos de Actualización de Histogramas

En esta sección explicaremos los mecanismos propuestos para actualizar la información dentro de un histograma. Se han diseñado cinco estrategias para trabajar con histogramas: min-max, adición de casillas, actualización de la frecuencia ponderada, límites dinámicos y fusión dinámica. Los tres primeros mecanismos se utilizan para los histogramas convencionales y los dos últimos están diseñados para ser aplicados sobre histogramas dinámicos.

1) Estrategia Min-max: Podemos enfrentarnos a la posibilidad de que no podamos manejar los datos y tengamos que procesarlos por lotes, pero conocemos los límites inferior y superior de nuestras variables. Si tenemos esta información, podemos hacer el histograma con estos límites dividiendo el espacio entre el máximo y el mínimo en trozos de igual tamaño. Este escenario puede no ser común, discutiremos metodologías más desinformadas en las próximas secciones. La actualización de las frecuencias se hará añadiendo los nuevos valores a las casillas correspondientes.

2) Adición de casillas: Con el primer lote de datos construiremos un histograma que servirá como base para las siguientes actualizaciones.

Cuando llegan nuevos datos como entrada pueden ocurrir tres cosas: tienen una casilla válida, poseen un valor menor que el mínimo o poseen un valor mayor que el máximo. Esto significa que, si un nuevo valor no tiene casilla asignable, sólo puede ser porque es menor o mayor que cualquier otro valor observado.

Esta estrategia trata de cubrir el espacio no visto hasta que se produzca una casilla válida para el nuevo dato. En la práctica lo que haremos es añadir tantas casillas a la izquierda o a la derecha como necesitemos para cubrir el nuevo valor. Esto añadirá nuevas casillas a nuestro histograma aumentando el número de las mismas fijado en la primera etapa.

3) Actualización de la frecuencia ponderada: Empleando el primer lote de datos creamos un histograma inicial. Cuando se introducen nuevos valores, el algoritmo construirá otro histograma con ellos. Con esto tendremos dos histogramas al mismo tiempo: el antiguo $histogram_{old}$ y el actual $histogram_{curr}$.

Construiremos un nuevo histograma con los valores de $histogram_{old}$ y $histogram_{curr}$. Este histograma tiene en cuenta el nuevo mínimo y máximo que puede haber cambiado. Teniendo el nuevo histograma vacío $histogram_{new}$ lo rellenamos de la siguiente manera.

Para cada una de las casillas del nuevo histograma, comprobaremos si el antiguo y el actual tienen casillas que se solapan con ésta. Si es así, la cantidad con la que va a contribuir cada histograma será proporcional a la anchura del solapamiento.

Sean los histogramas: $h_{old} = [h^0_{old}, ..., h^k_{old}]$, $h_{curr} = [h^0_{curr}, ..., h^k_{curr}]$ y $h_{new} = [h^0_{new}, ..., h^k_{new}]$. Para una casilla dada del nuevo histograma $[h^j, h^{j+1}]$ definimos la relación de solapamiento con una casilla de h_{curr} $[h^i_{curr}, h^{i+1}_{curr}]$ mediante la ecuación 1.

$$\begin{cases} \frac{h^{j+1}-h^i_{curr}}{h^{j+1}-h^j} & if\ h^{j+1} > h^i_{curr}\ and\ h^j < h^i_{curr}\ and\ h^{j+1} < h^{i+1}_{curr} \\ \frac{h^{j+1}-h^j}{h^{j+1}-h^j} & if\ h^j > h^i_{curr}\ and\ h^{j+1} < h^{i+1}_{curr} \\ \frac{h^{i+1}_{curr}-h^j}{h^{j+1}-h^j} & if\ h^j < h^{i+1}_{curr}\ and\ h^{j+1} > h^{i+1}_{curr} \end{cases} \quad (1)$$

Esta forma de medir el solapamiento nos da un número en el rango $[0, 1]$ para las casillas del histograma antiguo y del actual. Con este número tenemos la frecuencia ponderada que contribuirá a la creación del nuevo histograma.

4) Límites Dinámicos: El componente central de los histogramas dinámicos es la anchura de cada casilla, lo que nos indica la densidad de valores en ella. La idea principal de esta estrategia es preservar el porcentaje de anchura sobre el rango total de cada una de las casillas. Esto significa que si una casilla cubre un 10 % del rango de anchura total, cambiaremos los límites correspondientes de las casillas en las actualizaciones para preservar este 10 % de cobertura.

Para un histograma dado $h = [h_0, h_1, ..., h_k]$ calculamos los porcentajes de la forma $p = [\frac{h_1-h_0}{h_k-h_0}, \frac{h_2-h_1}{h_k-h_0}, ..., \frac{h_k-h_{k-1}}{h_k-h_0}]$.

Todos estos valores están en el rango $[0, 1]$ y suman 1. Esta estrategia crea un histograma la primera vez que entrenamos el modelo. Cuando el algoritmo reciba nuevos datos, actualizará los histogramas ya existentes. En cada época o paso de entrenamiento, calcularemos de nuevo los porcentajes de la anchura de las casillas para poder conservarlos al final del proceso de entrenamiento. Si aparecen valores repetidos en los nuevos datos, éstos se establecen en las casillas correspondientes para mantener la filosofía de los Histogramas Dinámicos.

Actualizaremos el máximo y mínimo si corresponde. Empezando por el mínimo calculamos los nuevos límites. El antiguo histograma será el h definido anteriormente y el nuevo histograma se notará como h_{new}. La posición i-ésima de este nuevo histograma se define como $h^i_{new} = h^{i-1}_{new} + p_{i-1} \cdot (max - min)$.

Donde max y min son el nuevo máximo y mínimo del histograma. Siguiendo este esquema completaremos los nuevos límites para las casillas terminando en el nuevo máximo y actualizando las frecuencias al final del proceso.

5) Fusión Dinámica: Este método combina dos histogramas dinámicos existentes. Para ello, contamos con un histograma ya existente y crearemos uno nuevo con los datos de entrada que se mezclará con el antiguo para actualizarlo.

La idea detrás de esta estrategia es preservar la densidad de los dos histogramas, es decir, la relación entre las frecuencias

y los porcentajes de anchura de las casillas. Después de construir el segundo histograma, calculamos la densidad media de ambos y creamos un nuevo histograma que conserva esta métrica.

Dado un histograma $h = [h_0, h_1, ..., h_k]$ entonces cada porcentaje se calcularía como $p_i = \frac{h_i - h_{i-1}}{h_k - h_0}$. Si el recuento de frecuencias del histograma es $c = [c_1, ..., c_k]$, la densidad de cada casilla sería $d_i = \frac{c_i}{p_i}$.

Ahora tenemos dos listas de densidades por lo que podemos calcular la media de cada una de ellas y luego la media de esos dos valores terminando con una densidad media que anotaremos como D_M.

Para cada paso h^i_{new} se calculará como $h^i_{new} = n^{i-1}_{new} + p^i_{new} \cdot (max - min)$ donde max y min son los nuevos límites inferior y superior del histograma y p^i_{new} es el porcentaje de la casilla correspondiente. Hemos calculado la densidad como $densidad = \frac{frecuencia}{porcentaje}$ por lo que ahora podemos resolver que $porcentaje = \frac{frecuencia}{densidad}$. Conocemos la densidad media D_M y la frecuencia de cada casilla en el nuevo histograma, por lo que podemos calcular el porcentaje de cada casilla en el nuevo histograma. Siguiendo este proceso terminaremos los límites del nuevo histograma y acabaremos colocando como límite superior el nuevo máximo.

III-B. VAMBH: Valoración de Anomalías Multi-paso Basada en Histogramas

VAMBH es un algoritmo de detección de anomalías no supervisado que aborda los dos inconvenientes mencionados de los métodos basados en histogramas. También hace uso del mecanismo HBOS para generar las puntuaciones de anomalías de forma eficiente. Este algoritmo es capaz de manejar mayores cargas de datos en comparación con HBOS, y de actualizar la información de los histogramas.

El algoritmo itera sobre los datos por lotes. Esto hará que se cree un primer estado interno a partir de los datos de entrada. Después de este primer ajuste del modelo, se llamará iterativamente a una de las estrategias anteriores con nuevos datos, actualizando el modelo interno. Esta forma de proceder es el diseño principal de VAMBH. Ajustar el algoritmo varias veces se traduce en la capacidad de VAMBH de actualizar y mejorar el modelo interno. Por ello este método es capaz, no sólo de ingerir mayores cargas de datos a trozos en comparación con otros métodos, sino también de tener un conocimiento más general de los datos de entrada.

El algoritmo crea un primer histograma. Tras este proceso, cada vez que lleguen nuevos datos al algoritmo, se llamará al mecanismo de actualización correspondiente. Esta rutina de entrenamiento terminará con un histograma que contiene el conocimiento general de todos los lotes de datos pasados al algoritmo. El procedimiento de evaluación será el mismo que el utilizado con HBOS con la función logaritmo.

IV. Marco experimental, resultados y análisis

En esta sección, detallaremos la metodología que hemos seguido para validar la propuesta VAMBH. El objetivo es probar la validez de VAMBH frente a HBOS, midiendo el rendimiento en una variedad de conjuntos de datos, así como explicar el comportamiento del algoritmo y estudiar las diferencias con los comparados. El repositorio de conjuntos de datos empleado ha sido la biblioteca Outlier Detection Datasets [2] junto con un conjunto de ingeniería real cedido por ArcelorMittal.

La Sección IV-A describe el marco experimental, en el que explicaremos los conjuntos de datos, así como los algoritmos involucrados y la metodología seguida para optimizar los parámetros. En la Sección IV-B se detallan los resultados experimentales y la discusión y análisis derivados de los mismos.

IV-A. Marco Experimental

En esta sección se explica la configuración de los experimentos. En primer lugar, se ofrece una descripción de los conjuntos de datos, seguida de los algoritmos utilizados, la optimización de sus hiperparámetros y las métricas de validación.

Conjuntos de datos: ODDS es un repositorio de conjuntos de datos etiquetados para la tarea de detección de anomalías. Hemos utilizado 23 conjuntos de datos de esta biblioteca, con una amplia gama de número de instancias y características para comprobar todos los aspectos de la propuesta.

Los conjuntos de datos tienen un porcentaje de anomalías que varía entre $1{,}2\,\%$ y $36\,\%$. El número de instancias de estos 23 conjuntos oscila entre 129 y $49,097$. En cuanto al número de características, va desde 6 hasta 400. Estos 23 conjuntos de datos nos darán suficientes casos para estudiar el rendimiento de forma general.

El problema provisto por ArcelorMittal es un problema de mantenimiento predictivo. El conjunto de datos tiene los registros de los sensores de una máquina de producción. Esta máquina tiene más de 100 sensores con registros de casi dos años, lo que nos da casi 40 millones de muestras. Este conjunto de datos representa una tarea muy dura para los algoritmos clásicos que normalmente tratan con una cantidad menor de muestras.

El conjunto de datos está débilmente etiquetado, con etiquetas temporales en los eventos de mantenimiento. Un mantenimiento es un problema grave que abarca un gran espacio de tiempo y requiere que la máquina deje de funcionar. El objetivo es detectar la causa de los mantenimientos antes de que se produzcan, ya que son los eventos que detienen la máquina.

Algoritmos y optimización: Comparamos con el algoritmo base HBOS disponible en la librería PyOD [7].

Para evaluar los algoritmos en un escenario justo, se ha aplicado un procedimiento de optimización a cada uno de ellos empleando Optuna [6]. Se han dado 100 combinaciones de parámetros a cada algoritmo por parámetro libre.

En el caso de VAMBH, se realiza un particionamiento de los conjuntos de datos de ODDS. De esta forma podemos entrenar el algoritmo por lotes.

Para evaluar el rendimiento en el conjunto de ArcelorMittal una ventana deslizante recorre los datos y comprueba si las puntuaciones de anomalías son mayores que un umbral. El tamaño de esta ventana deslizante es de dos horas, ya que es un intervalo de tiempo adecuado según los expertos en maquinaria de ArcelorMittal. Si es así, la muestra se marca como anomalía. Si se supera un determinado porcentaje de anomalías en la ventana deslizante, concluimos que tenemos una anomalía en dicha ventana. El comportamiento perfecto sería marcar todas las ventanas deslizantes en las seis horas anteriores a un mantenimiento como anómalas y el resto como normales.

Debemos tener en cuenta que se trata de un problema difícil, ya que está débilmente etiquetado. No disponemos de un oráculo sólido para evaluar cada muestra de forma independiente como anómala o normal. Para las pruebas, hemos utilizado los últimos 30 días del conjunto de datos para medir el rendimiento. En estos 30 días tenemos 25 mantenimientos y representan 2,5 millones de muestras en total.

***Métricas de rendimiento**:* Para evaluar un conjunto de datos en ODDS, el algoritmo lo procesa completamente, obteniendo puntuaciones para todas las muestras. Estas puntuaciones se utilizan para obtener las etiquetas con el fin de calcular posteriormente las métricas. Para cada conjunto de datos sabemos de antemano el número de anomalías, por lo que podemos utilizar este número para etiquetar exactamente las muestras empezando por las de mayor puntuación.

En el caso de ArcelorMittal para cada algoritmo, se ha elegido una cantidad adecuada de muestras para que puedan funcionar. VAMBH es capaz de entrenar con tantas muestras como se desee mientas que HBOS presenta limitaciones. Para la experimentación hemos limitado la selección de VAMBH a los primeros 30 días del conjunto de datos y de HBOS a las primeras 80 mil instancias.

Se han utilizado cinco métricas principales para probar el rendimiento de los algoritmos en los 23 conjuntos de datos: precisión, exhaustividad, puntuación F1, ROC-AUC y tiempo de cálculo utilizando los hiperparámetros optimizados para todos los conjuntos de datos.

Hemos añadido a las métricas el Área Bajo la Curva Característica Operativa del Receptor ya que es una métrica estándar en la clasificación binaria. El ROC-AUC no es la métrica ideal cuando se trata de la detección de anomalías, siendo la precisión y la exhaustividad mejores métricas para ese propósito. Esto se debe al promedio sobre todos los umbrales posibles que realiza el ROC-AUC. En este problema conocemos de antemano el número de anomalías para cada conjunto de datos, por lo que podemos calcular el umbral exacto para cada algoritmo en cada conjunto de datos.

IV-B. Resultados y Análisis

***Outlier Detection Datasets**:* El objetivo de estos experimentos es comprobar el funcionamiento de los algoritmos en un escenario de propósito general. Los conjuntos de datos tienen una gran variedad de número de muestras y características, por lo que pueden ofrecer una buena visión general del rendimiento.

Algoritmos	Prec.	Ex.	F1	AUC	T.(s)	TO(s)
HBOS	11.35	11.32	11.34	18.06	**2.81**	**842.61**
VAMBH	**14.23**	**12.47**	**12.47**	**18.40**	10.09	5,064.97

Tabla I: Resultados de los algoritmos agregando todos los conjuntos de datos. Mejor valor en negrita.

En la Tabla I hemos calculado para cada algoritmo las seis métricas en cada conjunto de datos y las hemos sumado para formar una métrica general para cada uno de los algoritmos.

De la Tabla I podemos concluir que VAMBH es mejor que HBOS en todas las métricas. En precisión el algoritmo está 3 puntos por encima de HBOS y en exhaustividad y F1 puntúa más de un punto por encima. Observando la puntuación F1 podemos ver que VAMBH es el algoritmo más equilibrado de los dos. Podemos ver también que superamos a HBOS en el área bajo la curva ROC. En cuanto al tiempo podemos ver que VAMBH consume algo más de tiempo que HBOS pero mejora el rendimiento sensiblemente.

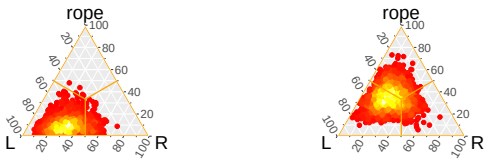
Hemos realizado una comparación más detallada entre VAMBH y HBOS en términos de puntuación F1 y ROC-AUC para todos los conjuntos de datos con el fin de evaluar las diferencias y similitudes en el rendimiento de estos algoritmos.

	F1		AUC	
Conjunto de datos	HBOS	VAMBH	HBOS	VAMBH
annthyroid	**0.7004**	0.6667	**0.9722**	0.9697
arrythmia	**0.5303**	0.5151	**0.8146**	0.8010
breastw	0.9373	**0.9498**	0.9850	**0.9936**
cardio	0.4943	**0.7841**	0.8283	**0.9365**
glass	0.1111	**0.4444**	0.7057	**0.7274**
ionosphere	0.3571	**0.5714**	0.5634	**0.6869**
letter	0.10	**0.18**	0.5842	**0.6370**
lympho	**0.8333**	0.6667	**0.9965**	0.2964
mammography	0.2808	**0.3769**	0.8534	**0.8740**
mnist	0.1571	**0.2571**	0.6224	**0.8557**
musk	**1**	**1**	**1**	**1**
optdigits	0.2933	**0.1133**	**0.8996**	0.6610
pendigits	0.3589	**0.5560**	**0.9354**	0.9130
pima	0.5458	**0.5560**	0.7074	**0.7078**
satellite	0.6537	**0.6719**	0.8108	**0.8340**
satimage-2	0.7606	**0.8592**	**0.9844**	0.9821
shuttle	**0.9687**	**0.9687**	0.9907	**0.9950**
speech	0.0491	**0.0656**	**0.4712**	0.4676
thyroid	**0.8280**	0.8065	**0.9922**	0.9910
vertebral	0.0333	**0.1**	0.3056	**0.5487**
vowels	0.18	**0.2**	0.6903	**0.6990**
wbc	0.6667	**0.7143**	0.9506	**0.9604**
wine	**0.5**	**0.5**	**0.8639**	0.8622
Media	0.4930	**0.5401**	**0.8055**	0.8

Tabla II: Resultados para cada conjunto de datos de ODDS para VAMBH y HBOS. Mejores resultados en negrita.

Como podemos ver en la Tabla II, VAMBH consigue un mejor rendimiento en general comparado con HBOS, obteniendo mejores resultados en la puntuación F1. En ROC-AUC se puede observar que ambos algoritmos empatan en promedio, pero VAMBH gana en más conjuntos de datos que HBOS (13 y un empate). En la puntuación F1, VAMBH supera a HBOS en 16 de 23 conjuntos de datos, empata en 3 y HBOS sólo gana en 4.

Para validar las diferencias de rendimiento entre HBOS y VAMBH, se ha realizado un test bayesiano de rango con signo [9] sobre las puntuaciones F1 y ROC-AUC obtenidas de los 23 conjuntos de datos para ambos algoritmos. Estos resultados se muestran en las Figuras 1a y 1b. En ambas figuras podemos ver que la mayoría de los puntos caen en el lado izquierdo del triángulo, siendo esta diferencia especialmente notable en las puntuaciones F1 señalándose que la diferencia es significativa.



(a) F1 (b) ROC-AUC

Figura 1: Tests bayesianos de rango con signo.

Conjunto de datos de ArcelorMittal: En esta sección describimos los resultados obtenidos por los algoritmos en el conjunto de datos de ArcelorMittal.

Algoritmos	Prec.	Ex.	F1	AUC	Muestras
HBOS	0.3785	**0.9406**	0.5398	0.5093	80K
VAMBH	**0.4931**	0.70	**0.5796**	**0.6354**	**2.5M**

Tabla III: Resultados sobre el conjunto de datos de ArcelorMittal. Mejores resultados en negrita.

La Tabla III muestra los resultados obtenidos por cada algoritmo. VAMBH obtiene el mejor resultado en precisión, siendo HBOS el mejor en exhaustividad. Sin embargo, observando la puntuación F1, VAMBH consigue el mejor rendimiento, siendo el algoritmo más equilibrado. VAMBH se mantiene 4 puntos por encima de HBOS en la puntuación F1. Esto sitúa a VAMBH como el algoritmo más equilibrado en términos de detección de TP y mitigación de FN y FP. Si miramos el ROC-AUC podemos ver que VAMBH está 13 puntos por encima de HBOS.

Las puntuaciones F1 y el ROC-AUC obtenidos no son muy altos para ninguno de los algoritmos. Esto es una consecuencia de la dificultad del problema expuesta en la sección anterior. El conjunto de datos está débilmente etiquetado, lo que da lugar a un problema de detección más difícil y a una evaluación complicada. No obstante, VAMBH ha mostrado mejores resultados en las métricas.

En cuanto al número de muestras que ingiere cada uno de los algoritmos, podemos ver que HBOS sólo puede lidiar con 80 mil instancias. En el caso de VAMBH, puede procesar datos sin limitaciones, pero a efectos de este estudio experimental hemos utilizado 2,5 millones de muestras.

V. Conclusiones

Este trabajo presenta un nuevo algoritmo llamado VAMBH que trabaja con histogramas, aplicándolo al problema de detección de anomalías. VAMBH se basa en varios métodos para actualizar la información dentro de un histograma, permitiendo actualizarlos al procesar los datos por lotes. Esta estrategia ha permitido ampliar y resolver los inconvenientes de uno de los mejores algoritmos basados en histogramas en la detección de anomalías: HBOS.

Esta nueva propuesta se ha aplicado al problema de detección de anomalías ampliando y resolviendo los inconvenientes de un algoritmo basado en histogramas. Como nota, nuestro método definido no sólo se aplica a este algoritmo conciso o a la tarea de detección de anomalías, sino que puede aplicarse a cualquier algoritmo de Aprendizaje Automático que utilice histogramas. Los mecanismos de actualización de histogramas diseñados permiten extender los algoritmos clásicos adaptándolos para trabajar con flujos de datos, consiguiendo potencialmente una mejora de rendimiento respecto a su diseño base.

Acknowledgment

Este trabajo ha sido parcialmente apoyado por el Ministerio de Ciencia y Tecnología bajo el proyecto TIN2017-89517-P, el Contrato UGR-AM OTRI-4260 y el proyecto de Excelencia Andaluza P18-FR-4961. J. Carrasco ha contado con el apoyo del Ministerio de Ciencia en el marco del Programa FPU 998758-2016.

Referencias

[1] Goldstein, M., & Dengel, A. (2012). Histogram-based outlier score (hbos): A fast unsupervised anomaly detection algorithm. KI-2012: Poster and Demo Track, 59-63.

[2] Shebuti Rayana (2016). ODDS Library [http://odds.cs.stonybrook.edu]. Stony Brook, NY: Stony Brook University, Department of Computer Science.

[3] Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. ACM computing surveys (CSUR), 41(3), 1-58.

[4] Aggarwal, C. C. (2015). Outlier analysis. In Data mining (pp. 237-263). Springer, Cham.

[5] Pevný, T. (2016). Loda: Lightweight on-line detector of anomalies. Machine Learning, 102(2), 275-304.

[6] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta,and Masanori Koyama. 2019. Optuna: A Next-generation Hyperparameter Optimization Framework. In KDD.

[7] Zhao, Y., Nasrullah, Z. and Li, Z., 2019. PyOD: A Python Toolbox for Scalable Outlier Detection. JMLR, 20(96), pp.1-7.

[8] Ramotsoela, D., Abu-Mahfouz, A., & Hancke, G. (2018). A survey of anomaly detection in industrial wireless sensor networks with critical water system infrastructure as a case study. Sensors, 18(8), 2491.

[9] Carrasco, J., García, S., del Mar Rueda, M., & Herrera, F. (2017, June). rNPBST: an R package covering non-parametric and bayesian statistical tests. In International Conference on HAIS 2017 (pp. 281-292). Springer, Cham.

[10] Zamini, M., & Hasheminejad, S. M. H. (2019). A comprehensive survey of anomaly detection in banking, wireless sensor networks, social networks, and healthcare. IDT, 13(2), 229-270.

[11] Fernandes, G., Rodrigues, J. J., Carvalho, L. F., Al-Muhtadi, J. F., & Proença, M. L. (2019). A comprehensive survey on network anomaly detection. Telecommunication Systems, 70(3), 447-489.

[12] Ahmad, S., Lavin, A., Purdy, S., & Agha, Z. (2017). Unsupervised real-time anomaly detection for streaming data. Neurocomputing, 262, 134-147.

[13] Moustafa, N., Hu, J., & Slay, J. (2019). A holistic review of network anomaly detection systems: A comprehensive survey. JNCA, 128, 33-55.

[14] Fahim, M., & Sillitti, A. (2019). Anomaly detection, analysis and prediction techniques in iot environment: A systematic literature review. IEEE Access, 7, 81664-81681.

Estudio experimental sobre XGBoost, Rotation Forest y su combinación

Francisco Javier González-Moya
Dpto. Ingeniería Informática
Universidad de Burgos
Burgos, España
fjgmoya@ubu.es

Álvar Arnaiz-González
Dpto. Ingeniería Informática
Universidad de Burgos
Burgos, España
alvarag@ubu.es

Juan José Rodríguez Díez
Dpto. Ingeniería Informática
Universidad de Burgos
Burgos, España
jjrodriguez@ubu.es

***Resumen*—Los ensembles son algoritmos multiclasificador que en general proporcionan mejores resultados predictivos que los obtenidos por clasificadores individuales. Rotation Forest y XGBoost son dos ensembles que han demostrado muy buenos resultados en diversos ámbitos. Este artículo compara los resultados de XGBoost y Rotation Forest, así como la combinación de ambos usando XGBoost como clasificador base de Rotation Forest.**

Se ha realizado un análisis experimental con diversos conjuntos de datos y dos implementaciones de los métodos utilizando rankings y test bayesianos para su comparación. En general, se mejora la precisión aplicando Rotation Forest sobre XGBoost en comparación con la obtenida por XGBoost de manera independiente. Respecto del tiempo de ejecución, los resultados son distintos en función de la implementación, pero en todo caso inferiores al tiempo consumido por Rotation Forest.

***Index Terms*—Rotation Forest, XGBoost, ensembles, PCA, diversidad**

I. Introducción

Los ensembles son conjuntos de clasificadores que cooperan para obtener un rendimiento más favorable que aquel que se obtiene por un único clasificador funcionando de manera aislada [12]. La "sindicación" de predicciones de los clasificadores individuales ofrece en general mejores resultados. Algunos trabajos recientes [9] ofrecen resultados en esta dirección. La idea de disponer de subconjuntos de datos que entrenen de manera sesgada a cada clasificador ofrecen una mejora de la predicción [11], [14].

XGBoost [4] es un conocido algoritmo de gradient boosting [8]. Su técnica mejora la predicción por sucesivas iteraciones de un clasificador inicialmente poco preciso. Genera diversos árboles de decisión, pudiendo emplear distintos atributos en la construcción de cada subárbol.

Rotation Forest [15] es un método de creación de ensembles, que emplea el algoritmo PCA para proyectar los datos de entrenamiento y obtener un conjunto de datos rotado, orientado en la dirección de la primera componente principal. El entrenamiento del clasificador base (árbol de decisión por defecto) sobre el conjunto de datos rotado por PCA ofrecen a Rotation Forest mayor diversidad, lo que le otorga mayor capacidad predictiva que el uso del árbol de decisión de manera aislada [2]. La arquitectura del algoritmo permite sustituir el árbol de decisión por cualquier otro clasificador; también por uno de gradient boosting como XGBoost. Tanto Rotation Forest como XGBoost han demostrado ofrecer buen rendimiento predictivo.

Este artículo propone combinar Rotation Forest y XGBoost, aportando algunos resultados experimentales del mismo sobre precisión y tiempo de respuesta. Analizaremos además diferentes comportamientos según se utilice una implementación en Python [1], [13], [16] o en Java, mediante el uso de la biblioteca WEKA [7] de los algoritmos mencionados.

El documento se distribuye de la siguiente manera. El apartado II presenta la idea principal de combinar Rotation Forest con XGBoost y los objetivos que se desean conseguir. En el apartado III se detalla el conjunto de datasets empleados en los distintos experimentos, el diseño de los experimentos y las herramientas analíticas y gráficas que se emplearán en la comparación de resultados. El apartado IV detalla los resultados obtenidos, de manera general y particular, así como los test gráficos que comprueban por pares de experimentos la mejor opción. Finalmente, el apartado V presenta las principales conclusiones y esboza las líneas futuras de investigación.

II. Combinando XGBoost y Rotation Forest

Nuestra propuesta es incorporar rotaciones sobre XGBoost. De esta manera, se puede comparar su grado de precisión con el que aportan por separado tanto XGBoost como Rotation Forest. Se va a sustituir el clasificador base de Rotation Forest (Decision Tree en scikit-learn, j48 en la biblioteca WEKA) por XGBoost. Por lo tanto, se crea un ensemble de ensembles. Esto tiene sentido, dado que aportamos un clasificador base concebido como un conjunto de árboles que se mejoran progresivamente (boosting) y que ahonda en la idea de aportar diversos clasificadores para la predicción.

Cabe preguntarse en este punto cómo afectarán las rotaciones a la precisión de XGBoost, dado que en realidad el conjunto de datos que entrena cada árbol no se corresponde con un subconjunto de instancias originales, sino con el conjunto de los resultados obtenidos tras aplicar ciertas rotaciones.

Otra cuestión que se plantea de forma inmediata es el hecho de que los árboles o clasificadores base de Rotation Forest son

Este trabajo ha sido financiado bajo el proyecto BU055P20 (JCyL/FEDER, UE) de la Junta de Castilla y León, España, cofinanciado por fondos FEDER de la Unión Europea.

clasificadores independientes; mientras que XGBoost es (en sí mismo) un ensemble que aporta una decisión por consenso de varios árboles creados intencionadamente de forma sesgada, dado que emplea cada uno un subconjunto de instancias y un subconjunto de atributos.

Este estudio pretende obtener la respuesta a dos preguntas fundamentales: (i) ¿mejora la precisión de XGBoost gracias a su uso conjunto con Rotation Forest?, (ii) ¿afecta significativamente los tiempos de ejecución de la nueva propuesta? El objetivo del artículo es responder a estas preguntas mediante la experimentación.

III. Experimentación

Comenzamos detallando el conjunto de datasets empleados en toda la experimentación (apartado III-A). En el siguiente, apartado III-B, se indica brevemente cómo se ha realizado el diseño de los experimentos llevados a cabo y alguna justificación al respecto, en particular a cuestiones sobre el modo de abordar las rotaciones. El apartado III-C aborda brevemente el método de comparación analítica entre los distintos casos para poder ordenarlos de mejor a peor. Finalmente el apartado III-D detalla la construcción de los test estadísticos con apoyo en una herramienta gráfica, estableciendo la comparación de interés, dos a dos, entre las distintas configuraciones probadas.

III-A. Datasets empleados

La tabla I muestra un resumen de las principales características de los 104 conjuntos de datos empleados en los experimentos. Se trata de datasets bien conocidos, pertenecientes en su mayor parte al repositorio UCI [6] y ampliamente utilizados en otros experimentos aportados en la bibliografía de referencia [15].

III-B. Diseño de experimentos

Se han diseñado inicialmente tres tipos de experimentos. Los dos primeros consisten en ejecutar los clasificadores sin modificación: XGBoost (XGB) y Rotation Forest (RotF), siendo el tercero la combinación de Rotation Forest y XGBoost. En todos los casos, combinando un total de 1 000 iteraciones. En el tercer experimento el clasificador base (Decision Tree en scikit-learn, (DcT), o bien j48 en la biblioteca WEKA) se sustituye por el propio XGBoost, que hace la función de clasificador base.

Aplicamos un principio de equidad en el número de iteraciones en todos los experimentos. En el caso de nuestra propuesta de (RotF+XGB) debe tenerse en cuenta el número de iteraciones de Rotation Forest I_{RotF} y el número de iteraciones de XGBoost I_{XGB}, cuyo producto nos ofrece el número total de iteraciones. El producto sea siempre el mismo número de iteraciones en todos los casos.

Los resultados seguramente no serán los mismos si tenemos un número elevado de iteraciones de Rotation Forest I_{RotF} y acaso muchas menos en XGB I_{XGB}, y viceversa. Parece interesante desdoblar los experimentos con la propuesta de este artículo en dos configuraciones: la primera de ellas con $\mathrm{I}_{RotF} = 100$ iteraciones de Rotation Forest y $\mathrm{I}_{XGB} = 10$ iteraciones de XGB, y la segunda con $\mathrm{I}_{RotF} = 10$ y con $\mathrm{I}_{XGB} = 100$ respectivamente. Conseguimos cuatro configuraciones: XGB, RotF, RotF+XGB (100×10) y RotF+XGB (10×100). Las cuatro se ejecutaron en dos entornos: la primera en Python sobre un código previo [1], [13] y la segunda en Java empleando la biblioteca WEKA [7].

III-C. Ranking promedio

Emplearemos el porcentaje de victorias y el ranking promedio [5] como indicadores de la bondad de la predicción de cada configuración. Para cada dataset, se ordena la precisión obtenida por cada configuración y se asigna una puntuación desde uno al número de configuraciones, de manera que las mejores precisiones obtengan rankings más bajos. En caso de empate entre dos o más puntuaciones, se reparten los puntos en discusión entre las configuraciones que producen el empate. Finalmente, se calcula para cada configuración la media de puntuaciones, que es el ranking promedio, y se ordenan las configuraciones de menor a mayor promedio.

Este ranking toma en consideración no sólo a la configuración mejor clasificada respecto de cada dataset, sino que tiene en cuenta todas las puntuaciones obtenidas, y que se resumen con una media. Este es un indicador de una mejor "bondad en promedio" de cada configuración, respecto de la precisión obtenida.

En general, si se dispone de N configuraciones y se reparten N puntos enumerados desde uno, la suma de puntuaciones ofrece $N * (N + 1)/2$ siendo su media $(N + 1)/2$. Podemos decir que cualquier ranking promedio inferior a este punto medio indica un mejor comportamiento del experimento que en caso contrario. Para nuestro caso particular, al disponer de ocho configuraciones, el punto o valor medio sería 4.5.

III-D. Tests bayesianos

Se utilizaron los tests estadísticos bayesianos incorporando una representación gráfica [3], gracias a una librería en Python [10]. Esto nos permite establecer si las diferencias son o no significativas en comparaciones uno a uno entre configuraciones.

Cada test es un triángulo equilátero, al que se asigna a los vértices inferiores (denominados L y R) la representación de cada uno de los dos experimentos. El vértice superior o ROPE (*region of practical equivalence* o área de equivalencia práctica) delimita la zona donde los resultados así representados son indiferentes a los métodos comparados; es decir, no hay una ventaja en el uso de uno u otro. Se representa una nube de puntos que se distribuyen según una medida (en nuestro caso, precisión). Cada vértice del triángulo incorpora el tanto por uno de puntos de la muestra "mejores" respecto de su eje. Es necesario establecer un umbral de tolerancia máxima para las diferencias de precisión entre ambos métodos que incide en la consideración de "equivalencia práctica"; en nuestro caso hemos establecido este umbral en el 1 %.

IV. Resultados

A continuación se presenta de forma resumida los resultados obtenidos de los distintos experimentos. Se ofrece la tabla

dataset	inst.	atrib.	clases
abalone	4177	8	3
acute-inflammation	120	6	2
acute-nephritis	120	6	2
annealing	898	31	5
appendicitis	106	7	2
balance	625	4	3
balance-scale	625	4	3
bands	365	19	2
bank	4521	16	2
blood	748	4	2
breast-cancer	286	9	2
breast-cancer-wisc	699	9	2
breast-cancer-wisc-diag	569	30	2
breast-cancer-wisc-prog	198	33	2
breast-tissue	106	9	6
bupa	345	6	2
car	1728	6	4
cardiotocography-10clases	2126	21	10
cardiotocography-3clases	2126	21	3
chess-krvkp	3196	36	2
coil2000	9822	85	2
congressional-voting	435	16	2
conn-bench-sonar-mines-rocks	208	60	2
conn-bench-vowel-deterding	990	11	11
contrac	1473	9	3
credit-approval	690	15	2
cylinder-bands	512	35	2
dermatology	366	34	6
echocardiogram	131	10	2
energy-y1	768	8	3
energy-y2	768	8	3
fertility	100	9	2
haberman-survival	306	3	2
hayes-roth	160	3	3
heart-cleveland	303	13	5
heart-hungarian	294	12	2
heart-va	200	12	5
hepatitis	155	19	2
hill-valley	1212	100	2
horse-colic	368	25	2
image-segmentation	2310	18	7
ionosphere	351	33	2
iris	150	4	3
libras	360	90	15
mammographic	961	5	2
marketing	6876	13	9
molec-biol-promoter	106	57	2
molec-biol-splice	3190	60	3
monks-1	556	6	2
monks-2	601	6	2
monks-3	554	6	2
mushroom	8124	21	2

dataset	inst	atrib	clases
musk-1	476	166	2
musk-2	6598	166	2
newthyroid	215	5	3
oocytes-merluccius-nucleus-4d	1022	41	2
oocytes-merluccius-states-2f	1022	25	3
oocytes-trisopterus-nucleus-2f	912	25	2
oocytes-trisopterus-states-5b	912	32	3
optical	5620	62	10
ozone	2536	72	2
page-blocks	5473	10	5
parkinsons	195	22	2
phoneme	5404	5	2
pima	768	8	2
pittsburg-bridges-material	106	7	3
pittsburg-bridges-rel-l	103	7	3
pittsburg-bridges-span	92	7	3
pittsburg-bridges-t-or-d	102	7	2
pittsburg-bridges-type	105	7	6
planning	182	12	2
primary-tumor	330	17	15
ringnorm	7400	20	2
satimage	6435	36	6
seeds	210	7	3
semeion	1593	256	10
soybean	683	35	18
spambase	4601	57	2
spect	265	22	2
spectf	267	44	2
statlog-australian-credit	690	14	2
statlog-german-credit	1000	24	2
statlog-heart	270	13	2
statlog-image	2310	18	7
statlog-landsat	6435	36	6
statlog-vehicle	846	18	4
steel-plates	1941	27	7
synthetic-control	600	60	6
tae	151	5	3
teaching	151	5	3
texture	5500	40	11
thyroid	7200	21	3
tic-tac-toe	958	9	2
titanic	2201	3	2
twonorm	7400	20	2
vehicle	846	18	4
vertebral-column-2clases	310	6	2
vertebral-column-3clases	310	6	3
wall-following	5456	24	4
waveform	5000	21	3
waveform-noise	5000	40	3
wdbc	569	30	2
wine	178	13	3
wine-quality-red	1599	11	6

Cuadro I

DESCRIPCIÓN DE LOS CONJUNTOS DE DATOS UTILIZADOS, POR CADA UNO SE MUESTRA SU NOMBRE, NÚMERO DE INSTANCIAS, NÚMERO DE ATRIBUTOS Y NÚMERO DE CLASES.

general de ranking promedio y porcentaje de victorias de las ocho configuraciones, como aparece en la tabla II. Una explicación más detallada para cada uno de los entornos se resume en la tabla III para el entorno Python y en la tabla IV para las configuraciones en entorno Java. Finalmente, se presenta un resumen de los tiempos de respuesta en la figura 3 y un análisis de los mismos.

Algoritmo	rank.	% vic.
RotF+XGB 10×100 (P)	3.609	22.32
RotF+j48 1 000 (J)	3.842	8.54
RotF+XGB 100×10 (P)	4.059	21.33
RotF+DcT 1 000 (P)	4.064	24.13
XGB 1 000 (P)	4.510	14.23
RotF+XGB 100×10 (J)	4.629	4.25
RotF+XGB 10×100 (J)	4.827	3.09
XGB 1 000 (J)	6.460	2.10

Cuadro II

RESULTADOS DE RANKING PROMEDIO Y PORCENTAJE DE VICTORIAS EN ALGORITMOS (P)YTHON Y (J)AVA EN PLATAFORMA WEKA.

IV-A. Ranking promedio

En general, las implementación en Python de Rotation Forest + XGBoost y el uso de Rotation Forest por separado ofrecen un mejor comportamiento en promedio que el uso de XGB de forma independiente. Esto se observa claramente en la tabla II, tanto si se observa el ranking promedio como el porcentaje de victorias. Los algoritmos de rotación 100×10 y 10×100 proporcionan resultados similares entre sí, tanto si se observa en entorno Python como en Java. Los casos de Rotation Forest original se alternan en segundo y cuarto puesto de la tabla. La versión Python de Rotation Forest original se asemeja mucho a la versión Python de RotF+XGB 100×10.

Algoritmo	rank.	% vic.
RotF+XGB 10×100	2.262	26.57
RotF+XGB 100×10	2.475	27.06
RotF+DcT 1 000	2.545	26.24
XGB 1 000	2.718	20.13

Cuadro III
RESULTADO DEL RANKING PROMEDIO Y % DE VICTORIAS EN ENTORNO PYTHON.

Algoritmo	rank.	% vic.
RotF+j48 1 000	1.748	48.27
RotF+XGB 100×10	2.337	22.52
RotF+XGB 10×100	2.436	21.04
XGB 1 000	3.480	8.17

Cuadro IV
RESULTADO DEL RANKING PROMEDIO Y % DE VICTORIAS EN ENTORNO JAVA.

En la tabla III relativa a los resultados de ranking promedio en entorno Python, se puede observar un comportamiento similar al caso general, favorable a los algoritmos de rotación. De nuevo se mejora el resultado de XGB independiente, y en este caso también el resultado particular de RotF+DcT. El indicador de porcentaje de victorias se encuentra en este caso más ajustado que en el caso general.

En relación a los experimentos del entorno Java (WEKA) en la tabla IV, las configuraciones de rotación sobre XGB ofrecen de nuevo un mejor resultado que el obtenido sobre XGB independiente, tanto en ranking promedio como en porcentaje de victorias. Aunque en este caso el mejor resultado del grupo se obtiene utiliando RotF+j48 con amplia diferencia sobre todos los anteriores.

IV-B. Test bayesianos

En las figuras 1 y 2 se muestra la representación de los test bayesianos, en los cuales se pueden apreciar si las diferencias entre las configuraciones son significativas o no.

La figura 1 presenta los gráficos de los test bayesianos de Python. Las figuras (a) y (b) representan XGB contra RotF+XGB en sus dos configuraciones (100×10 y 10×100) e ilustran la mejoría del uso de las rotaciones en XGBoost respecto del uso de XGBoost de forma independiente. Resulta evidente, en estos casos, la mejoría derivada del uso de rotaciones con XGBoost. En el caso de los gráficos (c) y (d), que se refiere a Rot+DcT respecto de las configuraciones con rotación de XGB, no hay una opción preferible sobre otra porque ambos son equivalentes.

La figura 2 muestra la representación gráfica de los test bayesianos para las configuraciones en Java. Las figuras (a) y (b) representan de nuevo XGB contra RotF+XGB en sus dos configuraciones (100×10 y 10×100) e ilustran la mejoría del uso de las rotaciones respecto del uso de XGBoost de forma independiente. En este caso, la nube de puntos avanza un poco más sobre la zona de equivalencia práctica. Las ilustraciones (c) y (d) acentúan la indiferencia respecto al uso de RotF+j48 comparado con las rotaciones sobre XGB.

En este caso, los resultados de los gráficos de la figura 2 confirman con matices lo observado en el apartado anterior. El número de rotaciones de Rotation Forest en (a) y (b) parece más relevante que en el caso anterior; por otro lado, el rendimiento en (c) y (d) de RotF+j48 o de RotF+XGB (en sus dos configuraciones) ofrecen resultados equivalentes.

IV-C. Tiempo de ejecución

Otro objetivo, al margen de la precisión, es obtener una medida de tiempos de ejecución en cada uno de los escenarios, que permita evaluar la conveniencia de cada una de las soluciones respecto de la rapidez con la que puede ser obtenida.

En general, los tiempos de ejecución de los algoritmos implementados en Python son superiores a sus librerías homólogas en Java, aunque el caso 100×10 no siga esta pauta. En cada entorno, la comparación con la referencia que ofrece RotF ofrece un ahorro considerable en tiempo de ejecución, tanto en el caso de XGB como también en el caso de las configuraciones de rotación, como muestra la figura 3.

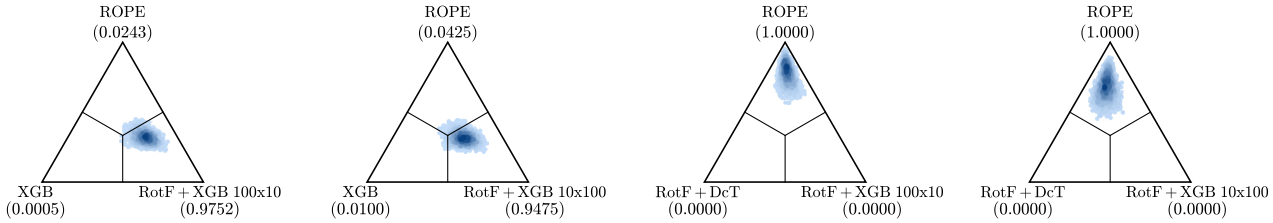
En el entorno Python, el tiempo consumido por la rotación más costosa, 100×10, es siete veces más rápida que la de RotF, aportando además valores muy similares a los tiempos obtenidos por XGB. En el entorno Java, el tiempo consumido por la rotación más costosa, 100×10, es muy similar al de RotF.

V. CONCLUSIONES Y LÍNEAS FUTURAS

Con la intención de mejorar los resultados predictivos de XGBoost, se han aportado tres experimentos. El primero con XGBoost de manera independiente, el segundo utilizando Rotation Forest original y el tercero combinando Rotation Forest con XGBoost en dos configuraciones de rotación 100×10 y 10×100. Se han ejecutado los experimentos en dos entornos diferentes como Python y Java, y se ha puesto de manifiesto la mejora de resultados de predicción de utilizar RotF+XGB frente a utilizar XGB sin rotaciones.

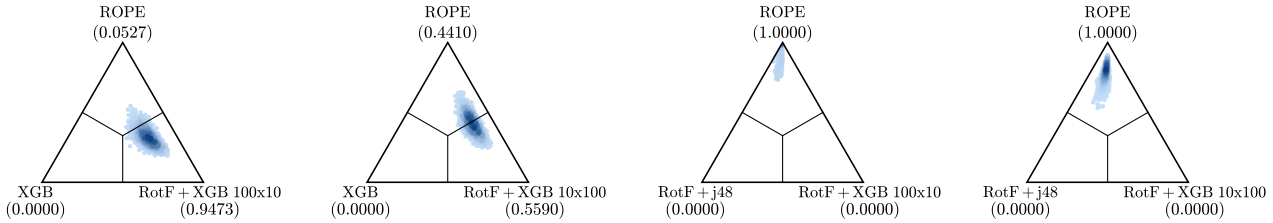
De acuerdo a la experimentación realizada, aplicar rotaciones a XGB mejora la precisión en general, hecho comprobado con las dos implementaciones. El uso de la configuración 100×10 ó 10×100 es menos relevante, de acuerdo a los test estadísticos en Python que en Java, respecto de la bondad del test. Los tiempos de ejecución son distintos por entornos, pero mejores que el tiempo empleado por RotF, dado que generamos un ensemble de menor tamaño. En entorno Python la diferencia en tiempo entre RotF y RotF+XGB 100×10 es (en promedio y en desviación estándar) siete veces mayor para el primero, mientras que en Java los tiempos en promedio son similares y la desviación estándar de una no llega a doblar a la otra.

Las diversas configuraciones de los algoritmos se han ejecutado bajo la premisa de parámetros por defecto, lo que en algunos casos puede no plantear una comparación equitativa. El clasificador base de RotF (DcT) actúa con libertad en cuanto a la máxima profundidad (`max_depth`) o altura del árbol, mientras que XGB por defecto emplea un máximo de seis niveles. Un ajuste de parámetros más fino puede aportar resultados diferentes a los aquí presentados.



(a) XGB vs RotF+XGB (b) XGB vs RotF+XGB (c) RotF+DcT vs RotF+XGB (d) RotF+DcT vs RotF+XGB

Figura 1. Test bayesianos de los experimentos en Python: XGB vs RotF+XGB (a, b) y RotF+DcT vs RotF+XGB (c y d)



(a) XGB vs RotF+XGB (b) XGB vs RotF+XGB (c) RotF+j48 vs RotF+XGB (d) Rot+j48 vs RotF+XGB

Figura 2. Test bayesianos de los experimentos en Java: XGB vs RotF+XGB (a, b) y RotF+j48 vs RotF+XGB (c y d)

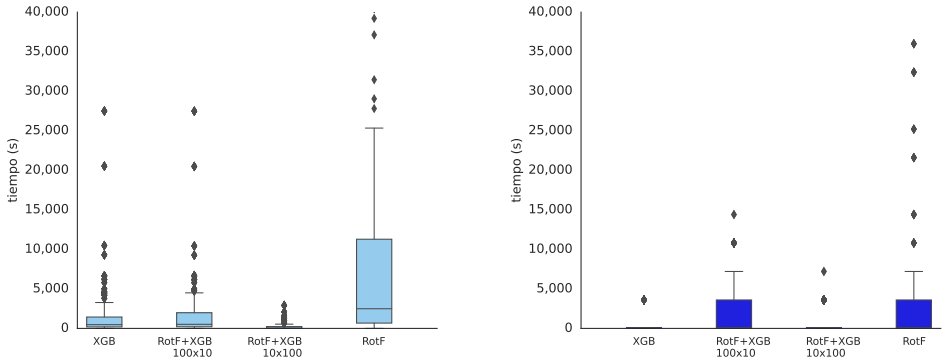


Figura 3. Diagrama de cajas con los tiempos de ejecución de las diversas configuraciones: en Python a la izquierda y en Java a la derecha.

Una línea de trabajo futuro podría ser la de probar nuevas implementaciones de algoritmos que se ejecuten en GPU, total o parcialmente. En especial para ejecuciones del algoritmo PCA y respecto a la implementación de Python, aunque también referido al clasificador XGB. Puede ser también interesante explorar las implementaciones en entorno WEKA que puedan existir al respecto.

Referencias

[1] A. Bagnall. Timeseries classification - algorithms review, 2018.

[2] A. Bagnall, M. Flynn, J. Large, J. Line, A. Bostrom, and G. Cawley. Is rotation forest the best classifier for problems with continuous features? *arXiv*, 2020.

[3] Alessio Benavoli, Giorgio Corani, Janez Demsar, and Marco Zaffalon. Time for a change: a tutorial for comparing multiple classifiers through bayesian analysis, 2017.

[4] Tianqi Chen and Carlos Guestrin. Xgboost. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Aug 2016.

[5] J. Demšar. Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine Learning Research*, 7:1–30, 2006.

[6] Dheeru Dua and Casey Graff. UCI machine learning repository, 2017.

[7] Eibe Frank, Mark A Ḧall, and Ian HẄitten. *The WEKA WorkbenchȮnline Appendix for Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, Fourth Edition, 2016.

[8] Yoav Freund and Robert E. Schapire. A decision-theoretic generalization of on-line learningand an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, August 1997.

[9] Sergio González, Salvador García, Javier Del Ser, Lior Rokach, and Francisco Herrera. A practical tutorial on bagging and boosting based ensembles for machine learning: Algorithms, software tools, performance study, practical perspectives and opportunities. *Information Fusion*, 64:205–237, 2020.

[10] Mario Juez-Gil, Álvar Arnaiz-González, Juan J. Rodríguez, Carlos López-Nozal, and César García-Osorio. Rotation forest for big data. *Information Fusion*, 74:39–49, 2021.

[11] Bartosz Krawczyk, Leandro L. Minku, João Gama, Jerzy Stefanowski, and Michał Woźniak. Ensemble learning for data stream analysis: A survey. *Information Fusion*, 37:132–156, 2017.

[12] Ludmila I Kuncheva. *Combining pattern classifiers: methods and algorithms*. John Wiley & Sons, 2014.

[13] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay.

Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[14] R. Polikar. Ensemble based systems in decision making. *IEEE Circuits and Systems Magazine*, 6(3):21–45, 2006.

[15] J.J. Rodriguez, L.I. Kuncheva, and C.J. Alonso. Rotation forest: A new classifier ensemble method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(10):1619–1630, 2006.

[16] Guido Van Rossum and Fred L. Drake. *Python 3 Reference Manual*. CreateSpace, Scotts Valley, CA, 2009.

Estudio de estrategias basadas en vecinos para clasificación multi-instancia multi-etiqueta

Eva Gibaja
Dept. Informática y Análisis Numérico
Universidad de Córdoba
Córdoba, España
egibaja@uco.es

Amelia Zafra
Dept. Informática y Análisis Numérico
Universidad de Córdoba
Córdoba, España
azafra@uco.es

***Resumen*—El aprendizaje multi-instancia multi-etiqueta extiende a los paradigmas de aprendizaje clásicos ofreciendo una gran versatilidad al utilizar una representación flexible tanto del espacio de entrada como del espacio de salida. En este trabajo proponemos un estudio de estrategias basadas en vecinos más cercanos en el que analizamos algoritmos que resuelven el problema haciendo una transformación de los datos a multi-etiqueta y algoritmos especializados que trabajan directamente con los datos multi-instancia multi-etiqueta y que han sido desarrollados en este trabajo. Los resultados muestran que tanto el método de transformación utilizado en los algoritmos que utilizan una transformación del problema, como la medida de distancia en los algoritmos especializados son elementos que influyen de forma directa en su rendimiento.**

***Index Terms*—multi-instancia, multi-etiqueta, no convencional, clasificación**

I. Introducción

La clasificación multi-instancia multi-etiqueta (MIML) es un paradigma de aprendizaje supervisado relativamente reciente propuesto en [1], que combina dos marcos de aprendizaje que destacan por su flexibilidad en la representación de los problemas a resolver. Por un lado, el aprendizaje multi-instancia (MI) permite una representación flexible del espacio de entrada asociando un patrón (*bolsa*) con múltiples instancias, todas ellas con el mismo número de atributos [2]. Por otro lado, el aprendizaje multi-etiqueta (ML) flexibiliza la representación del espacio de salida asociando a cada patrón no una, sino un conjunto de clases binarias (*etiquetas*) [3]. El aprendizaje MIML permite una representación natural de objetos complejos en problemas reales, tales como la clasificación de textos e imágenes, la anotación de video o imágenes o la bioinformática [4]. Por ejemplo, una imagen, como se muestra en la Figura 1, podría ser representada por una bolsa con múltiples instancias, representando cada una de ellas una región en la imagen, y cada imagen podría tener varias etiquetas binarias asociadas (en este caso: *puente*, *bosque*, *río*).

En [1] se describen dos enfoques para la resolución de problemas MIML (véase Figura 2). El primero consiste en transformar el problema a un problema MI o un problema ML y resolverlo con algoritmos diseñados para estos dos paradigmas de aprendizaje. El segundo consiste en desarrollar algoritmos específicos que abordan directamente el problema MIML, sin transformarlo previamente. En el caso de solu-

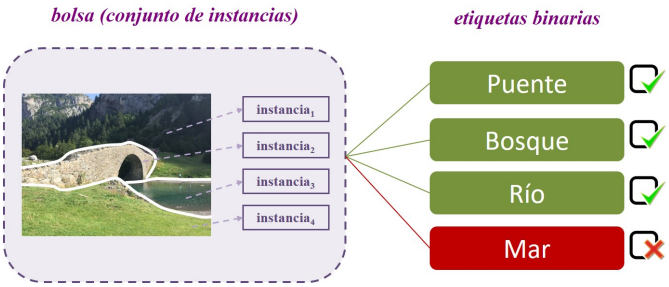


Figura 1. Representación MIML de una imagen

cionar un problema MIML mediante su transformación, se pueden aplicar dos vías o puentes. En una primera aproximación, que denominaremos *puente MI*, el problema MIML se transforma en un problema MI mediante alguna técnica de descomposición o agregación de etiquetas, y el problema resultante se resuelve utilizando cualquier algoritmo MI. Este enfoque se aplica sobre el espacio de salida, siendo Binary Relevance [5] y Label Powerset [5] dos transformaciones posibles. La segunda vía, que denominaremos *puente ML*, consiste en transformar el problema a una representación ML mediante alguna técnica que permita unificar las diferentes instancias de una bolsa en una única instancia, y luego solucionar el problema resultado mediante un algoritmo ML. Este enfoque se aplica en el espacio de entrada, siendo las transformaciones aritmética, geométrica y minimax algunas alternativas [6].

El rendimiento de los enfoques anteriores puede verse limitado debido a la pérdida de información incurrida durante el proceso de simplificación/transformación. Idealmente, las conexiones entre instancias y etiquetas, así como la correlación entre las propias etiquetas deberían ser tenidas en cuenta. Por este motivo, también se han propuesto técnicas especializadas que abordan el problema MIML de forma directa, sin hacer una transformación del problema. En la literatura se pueden encontrar propuestas basadas en redes neuronales [7], *ensembles* [8], máquinas de vector soporte [9] o vecinos más cercanos [10]. En [4] puede encontrarse una revisión más exhaustiva de algoritmos publicados en el aprendizaje MIML.

En este trabajo se lleva a cabo un estudio de diferentes alternativas para resolver el problema MIML utilizando algo-

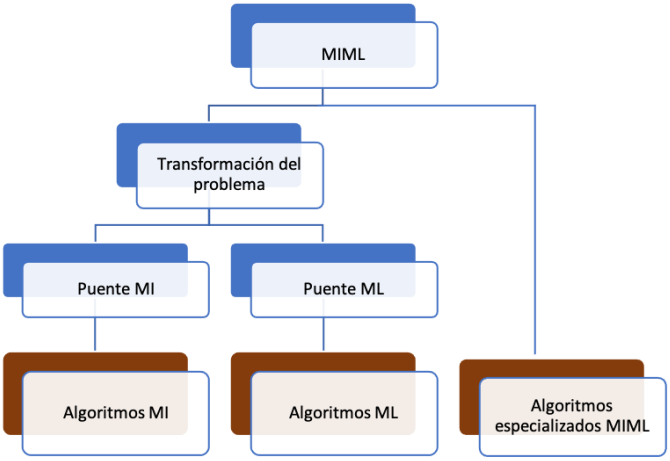


Figura 2. Algoritmos que resuelven el problema MIML

ritmos basados en los vecinos más cercanos (kNN, *k-nearest neighbours*). En el estudio se consideran dos alternativas: por un lado, la transformación del problema MIML mediante el puente ML y la posterior aplicación de algoritmos kNN para ML. Por otro lado, se han desarrollado algoritmos específicos para trabajar directamente sobre datos MIML basados en kNN. Para llevar a cabo este estudio, se analizan diferentes configuraciones de los algoritmos utilizando 8 conjuntos de datos diferentes y 7 métricas de evaluación.

El resto del artículo se estructura de la siguiente forma, la sección II define el problema MIML, a la vez que se describen los algoritmos basados en vecinos que se van a estudiar en este trabajo. En la sección III se describen los conjuntos de datos y las configuraciones utilizadas en la experimentación y se evalúan y discuten los resultados obtenidos. Finalmente, la sección IV presenta las conclusiones obtenidas.

II. Preliminares

II-A. *Definición formal del problema MIML*

Sean $\mathcal{X} = X_1 \times \cdots \times X_d$ un espacio de entrada d-dimensional, e $\mathcal{Y} = \{\lambda_1, \lambda_2, ..., \lambda_q\}$ un espacio de salida formado por un conjunto de q etiquetas. Sea $D = \{(X_i, Y_i) \mid 1 \leq i \leq m\}$ un dataset MIML de m bolsas, donde $X_i \subseteq \mathcal{X} = \{\mathbf{x}_{i1}, \mathbf{x}_{i2}, \ldots, \mathbf{x}_{in_i} | \ \mathbf{x}_{ij} \in \mathcal{X},\ j = 1, 2, \ldots, n_i\}$ es una bolsa formada por n_i instancias, e $Y_i \subseteq \mathcal{Y} = \{\lambda_1, \lambda_2, ..., \lambda_q \mid q > 1\}$ es el conjunto de etiquetas asociadas con la bolsa X_i, y el complemento de este conjunto, $\overline{Y}_i$, es el conjunto de etiquetas no relevantes. Cada subconjunto de $\mathcal{Y}$ se denomina *labelset*. Las asociaciones de etiquetas también se pueden representar mediante un vector binario $\mathbf{y} = (y_1, y_2, \ldots, y_q) = \{0, 1\}^q$ en el que cada elemento es 1 si la etiqueta es relevante o 0 en caso contrario. La clasificación MIML consiste en aprender una función $h_{\text{MIML}} : 2^{\mathcal{X}} \rightarrow 2^{\mathcal{Y}}$ que devuelve una bipartición del conjunto de etiquetas, $(Z_i, \overline{Z}_i)$, en relevantes y no relevantes.

II-B. *Algoritmos basados en vecinos más cercanos*

Los algoritmos basados en los vecinos más cercanos son ampliamente utilizados para realizar tareas de clasificación y se basan en predecir la clase de un nuevo ejemplo usando la clase a la que pertenecen los k ejemplos más cercanos (véase Algoritmo 1). Para determinar la cercanía se emplea una métrica de distancia o similitud que afecta directamente al rendimiento en este tipo de algoritmos.

En la siguientes secciones, se detallarán los algoritmos basados en kNN que se han utilizado en este estudio, diferenciando entre algoritmos kNN para ML (que requieren de la aplicación de un puente ML) y algoritmos kNN específicos para MIML (capaces de trabajar con los datos sin transformar).

Algoritmo 1 kNN

Entrada: $D = \{(\mathbf{x}_i, y_i) \mid 1 \leq i \leq m,\ \mathbf{x_i} \in \mathcal{X}, y_i \in \mathcal{Y}\}$
$\mathbf{x} = (x_1, \ldots, x_d)$ nuevo caso a clasificar
para $i = 1$ *to* m **hacer**
Calcular distancia $dist(\mathbf{x_i}, \mathbf{x})$
fin para
$D_{\mathbf{x}}^k = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_k, y_k)\} \subseteq D$ {los k casos ya clasificados más cercanos a $\mathbf{x}$}
devolver $\arg\max_{y \in \mathcal{Y}} \sum_{i=1}^{k} [\![y = y_i]\!]$ {la clase más frecuente en $D_{\mathbf{x}}^k$}

II-C. *Algoritmos ML basados en kNN*

En la literatura se pueden encontrar múltiples algoritmos basados en kNN para ML. BRkNN [11] es equivalente a aplicar de modo independiente kNN sobre cada etiqueta, pero calculando únicamente una vez los k vecinos más cercanos. La confianza en cada etiqueta se calcula como el porcentaje de vecinos que la incluyen. MLkNN [12] utiliza el principio MAP (*Maximum A Posteriori*) para hacer una predicción utilizando la información estadística sobre las etiquetas de los vecinos. DMLkNN [13] es una generalización de MLkNN en la que la regla MAP definida considera las dependencias entre etiquetas. IBLR_ML [14] combina aprendizaje basado en instancias y regresión logística. Tiene en cuenta las relaciones entre etiquetas utilizando las etiquetas de los vecinos como atributos extra en un esquema de regresión logística. MLDGC [15] se basa en el principio de la clasificación de datos gravitatoria (DGC), inspirado en los principios de la ley universal de Newton, para simular la atracción entre ejemplos. Considera cada ejemplo como una partícula de datos atómica e introduce el concepto de *Neighborhood-based Gravitation Coefficient* para utilizarlo en el cálculo de las fuerzas gravitatorias.

Estos algoritmos se pueden utilizar para resolver un problema MIML aplicando un puente ML que realiza una transformación previa del problema MIML a ML. De forma particular, en este trabajo se estudian tres métodos descritos en [6] que transforman cada bolsa en una única instancia:

- *Transformación aritmética (AT)*. El valor de cada atributo de la instancia resultado es el valor medio del atributo considerando todas las instancias de la bolsa.
- *Transformación geométrica (GT)*. El valor de cada atributo de la instancia resultado es el centro geométrico de

TABLA I
DATASETS MIML UTILIZADOS EN LOS EXPERIMENTOS

Dataset	Dominio	Número bolsas	Instancias/bolsa			atributos/bolsa	Número etiquetas	Cardinalidad	Densidad	Distinct
			media	*mínimo*	*máximo*					
birds	audio	282	7,400	1	36	38	19	2,010	0,105	125
3Sources	texto	416	2,278	1	3	4750	6	1,122	0,187	15
text	texto	2000	3,559	2	26	243	7	1,154	0,165	25
image	imagenes	2000	9,0	9	9	15	5	1,236	0,247	20
protein (subset 1)	proteinas	304	3,125	2	7	216	234	3,247	0,014	214
protein (subset 2)	proteinas	425	3,099	2	8	216	321	4,48	0,014	318
protein (subset 3)	proteinas	407	3,074	2	8	216	340	4,0	0,012	324
protein (subset 4)	proteinas	379	3,203	2	8	216	320	3,142	0,010	293

los valores máximos y mínimos considerando todas las instancias de la bolsa $(minimo + maximo)/2$.

- *Transformación Min-Max (MMT)*. Para cada atributo de la bolsa original, la instancia resultado incluye sus valores máximo y mínimo dentro de la bolsa, por lo que tendrá el doble de atributos de las instancias originales.

II-D. Algoritmos específicos MIML basados en kNN

En este trabajo presentamos como propuesta la adaptación de los algoritmos BRkNN, MLkNN, DMLkNN, IBLR_ML y MLDGC, descritos en la sección II-C, para resolver el problema MIML directamente, sin hacer una transformación previa del problema. En el proceso de adaptación de los algoritmos se ha mantenido su algorítmica en cuanto al tratamiento de las etiquetas del espacio de salida. Uno de los factores más influyentes en los algoritmos basados en kNN es la medida de distancia que determina la similitud entre los ejemplos. Para el caso MIML hemos adaptado la medida de distancia para trabajar con un espacio de entrada basado en bolsas que contienen múltiples instancias.

Dadas dos bolsas formadas por un conjunto variable de instancias, $A = \{a_1, a_2, \ldots, a_m\}$ y $B = \{b_1, b_2, \ldots, b_n\}$, se han utilizado las siguientes variantes de la distancia de Hausdorff:

- Maximal Hausdorff distance [16]:

$$MaxH(A,B) = \max\left\{\max_{a\in A}\min_{b\in B} dist(a,b), \max_{b\in B}\min_{a\in A} dist(b,a)\right\} \quad (1)$$

- Minimal Hausdorff distance [17]:

$$MinH(a,B) = \min_{a\in A, b\in B} dist(a,b) \quad (2)$$

- Average Hausdorff distance [18]:

$$AveH(A,B) = \frac{\sum_{a\in A}\min_{b\in B} dist(a,b) + \sum_{b\in B}\min_{a\in A} dist(b,a)}{|A| + |B|} \quad (3)$$

Donde *dist* mide la distancia entre dos instancias. En nuestro estudio se ha utilizado la distancia Euclídea.

III. EXPERIMENTOS

III-A. Descripción de los datos utilizados

Para este estudio, se han utilizado los siguientes 8 datasets:

- **text** [19]. Contiene datos de Reuters 21578. Se consideraron las 7 etiquetas más frecuentes. Las instancias en las bolsas adoptan la representación *Bag-of-Words* basada en la frecuencia de los términos que aparecen en el texto.
- **image** [20]. Contiene 2000 escenas de la naturaleza. Las etiquetas se añadieron de forma artificial a cada imagen. Cada imagen está representada como una bolsa de 9 instancias generadas por el método SBN [21].
- **protein** [22]. Contiene los datos genómicos de 7 tipos de organismos que cubren el denominado sistema de tres dominios. En este trabajo se han considerado los subconjuntos 1-4.
- **birds** [23]. Es un dataset para predecir la presencia de especies de pájaros, dado un audio mono con 10 segundos muestreado a 16kHz, 16 bits por segundo.
- **3sources** [24]. El dataset cubre 416 noticias del periodo Febrero–Abril 2009 que fueron recogidas de BBC, Reuters y The Guardian. Aparecían 169 noticias en las tres fuentes, 194 en dos fuentes, y 53 en una sola fuente. Cada noticia es una bolsa que puede contener de una a tres instancias y es descrita por la unión de los términos, en formato *tf*idf*, contenidos en cada una las fuentes que la reportaron. Cada noticia fue anotada manualmente.

La Tabla I muestra las principales características de estos datasets. La *cardinalidad* es el número medio de etiquetas por bolsa. La *densidad* es la cardinalidad dividida por el número total de etiquetas y se utiliza para poder comparar datasets con distinto número de etiquetas. Finalmente, *distinct* es el número de combinaciones de etiquetas diferentes en el dataset.

TABLA II
MÉTRICAS DE EVALUACIÓN

↑F-Measure	$2\frac{precision \cdot recall}{precision + recall}$	
Basadas en etiquetas		
	Macro-averaged	***Micro-averaged***
↑Precision	$\frac{1}{q}\sum_{i=1}^{q}\frac{tp_i}{tp_i + fp_i}$	$\frac{\sum_{i=1}^{q} tp_i}{\sum_{i=1}^{q} tp_i + \sum_{i=1}^{q} fp_i}$
↑Recall	$\frac{1}{q}\sum_{i=1}^{q}\frac{tp_i}{tp_i + fn_i}$	$\frac{\sum_{i=1}^{q} tp_i}{\sum_{i=1}^{q} tp_i + \sum_{i=1}^{q} fn_i}$
Basadas en ejemplos		
↓Hamming loss	$\frac{1}{m}\sum_{i=1}^{m}\frac{\mid Y_i \Delta\ Z_i \mid}{m}$	
↓Subset accuracy	$\frac{1}{m}\sum_{i=1}^{m}[\![Z_i = Y_i]\!]$	
↑Accuracy	$\frac{1}{m}\sum_{i=1}^{m}\frac{\mid Y_i \cap Z_i \mid}{\mid Y_i \cup Z_i \mid}$	
↑Precision	$\frac{1}{m}\sum_{i=1}^{m}\frac{\mid Y_i \cap Z_i \mid}{\mid Z_i \mid}$	
↑Recall	$\frac{1}{m}\sum_{i=1}^{m}\frac{\mid Y_i \cap Z_i \mid}{\mid Y_i \mid}$	
↓Ranking loss	$\frac{1}{t}\sum_{i=1}^{t}\frac{1}{\mid Y_i \mid \mid \overline{Y_i} \mid}\mid E \mid$	
	$E = \left\{(\lambda_a, \lambda_b) \mid \tau_i(\lambda_a) > \tau_i(\lambda_b), (\lambda_a, \lambda_b) \in Y_i \times \overline{Y_i}\right\}$	

TABLA III
RESULTADOS DE HAMMING LOSS ↓ Y RANKING MEDIO

Algoritmo	birds	3sources	image	text	protein01	protein02	protein03	protein04	Ranking
AT_BRkNN	0,0901	0,1584	0,1935	0,0632	0,0134	0,0143	0,0120	0,0100	2,2500 (2)
GT_BRkNN	0,0918	0,1206	0,1952	0,0685	0,0136	0,0142	0,0119	0,0099	2,2500 (2)
MMT_BRkNN	0,0905	0,1235	0,1902	0,0786	0,0133	0,0140	0,0118	0,0099	1,5000 (1)
AT_DMLkNN	0,0912	0,0874	0,1923	0,0551	0,0136	0,0139	0,0118	0,0098	1,6875 (1)
GT_DMLkNN	0,0933	0,0873	0,1962	0,0659	0,0136	0,0139	0,0118	0,0098	2,0620 (2)
MMT_DMLkNN	0,0925	0,0778	0,1863	0,0729	0,0136	0,0140	0,0118	0,0098	2,2500 (3)
AT_IBLR_ML	0,1080	0,0713	0,1925	0,0543	0,0381	0,0357	0,0351	0,0290	1,5000 (1)
GT_IBLR_ML	0,1073	0,0825	0,1939	0,0627	0,0390	0,0384	0,0350	0,0317	2,1250 (2)
MMT_IBLR_ML	0,1115	0,0694	0,1855	0,0699	0,0407	0,0407	0,0377	0,0312	2,3750 (3)
AT_MLkNN	0,0916	0,0878	0,1917	0,0549	0,0133	0,0138	0,0117	0,0097	1,8125 (2)
GT_MLkNN	0,0940	0,0833	0,1959	0,0656	0,0131	0,0139	0,0118	0,0098	2,5000 (3)
MMT_MLkNN	0,0920	0,0742	0,1869	0,0727	0,0135	0,0138	0,0117	0,0097	1,6875 (1)
AT_MLDGC	0,0873	0,1960	0,1927	0,0554	0,0135	0,0145	0,0119	0,0100	2,0000 (2)
GT_MLDGC	0,0899	0,1174	0,1947	0,0624	0,0136	0,0146	0,0122	0,0099	2,5000 (3)
MMT_MLDGC	0,0880	0,1283	0,1856	0,0714	0,0132	0,0141	0,0119	0,0098	1,5000 (1)
AveH_BRkNN	0,0817	0,2088	0,1837	0,0811	0,0131	0,0139	0,0118	0,0098	1,5000 (1)
MaxH_BRkNN	0,1000	0,2004	0,1915	0,0639	0,0141	0,0142	0,0119	0,0100	2,6250 (3)
MinH_BRkNN	0,0845	0,1943	0,1872	0,0842	0,0133	0,0139	0,0118	0,0098	1,8750 (2)
AveH_DMLkNN	0,0845	0,1126	0,1798	0,0741	0,0137	0,0139	0,0117	0,0098	1,2500 (1)
MaxH_DMLkNN	0,0983	0,1154	0,1902	0,0599	0,0139	0,0140	0,0118	0,0098	2,5000 (3)
MinH_DMLkNN	0,0888	0,1207	0,1882	0,0785	0,0137	0,0139	0,0118	0,0098	2,2500 (2)
AveH_IBLR_ML	0,1039	0,1035	0,1794	0,0748	0,0380	0,0399	0,0340	0,0272	2,1250 (2)
MaxH_IBLR_ML	0,1110	0,1082	0,1887	0,0603	0,0341	0,0390	0,0354	0,0270	2,2500 (3)
MinH_IBLR_ML	0,1080	0,0990	0,1864	0,0786	0,0357	0,0381	0,0339	0,0257	1,6250 (1)
AveH_MLkNN	0,0834	0,1126	0,1794	0,0742	0,0135	0,0138	0,0118	0,0096	1,6250 (1)
MaxH_MLkNN	0,0976	0,1122	0,1906	0,0601	0,0139	0,0139	0,0118	0,0098	2,6250 (3)
MinH_MLkNN	0,0869	0,1070	0,1889	0,0783	0,0134	0,0138	0,0117	0,0097	1,7500 (2)
AveH_MLDGC	0,0779	0,2109	0,1861	0,0778	0,0128	0,0135	0,0113	0,0096	1,5000 (1)
MaxH_MLDGC	0,1071	0,1968	0,1871	0,0595	0,0144	0,0145	0,0119	0,0101	2,5000 (3)
MinH_MLDGC	0,0815	0,1912	0,1886	0,0821	0,0129	0,0136	0,0108	0,0097	2,0000 (2)

III-B. Marco experimental

Se han estudiado 5 algoritmos basados en vecinos: BRkNN, MLkNN, DMLkNN, IBLR_ML y MLDGC. Todos los algoritmos se han ejecutado con 10 vecinos. En el caso de MLkNN y de DMLkNN *smooth* $= 1,0$. Se ha realizado una validación cruzada de 5 *fold*. Todos estos algoritmos y sus posibles configuraciones están implementados y disponibles en MIML (https://github.com/kdis-lab/MIML) [25], una librería Java para desarrollar, evaluar y comparar algoritmos de clasificación para MIML basada en Weka [26] y Mulan [27]. Los test estadísticos se han realizado con la librería R scmamp [28].

Para realizar el estudio, se han analizado 7 métricas de evaluación ML que incluyen métricas basadas en etiquetas (macro y micro F-measure) y en ejemplos (Hamming loss, subset accuracy, F-measure, accuracy y ranking loss). La Tabla II las define e indica con el símbolo ↑ que la métrica se maximiza y con el símbolo ↓ que la métrica se minimiza. En el caso de las métricas basadas en etiquetas, las expresiones tp_i, tn_i, fp_i y fn_i representan los valores de *true positive*, *false positive*, *false positive* y *false positive* para la etiqueta i. En el caso de las métricas basadas en ejemplos, para un predicado π, $[\![\pi]\!]$ devuelve 1 si el predicado es cierto y 0 en caso contrario y Δ representa la diferencia simétrica entre dos conjuntos de etiquetas (que se corresponde con el operador XOR de la lógica booleana). Si el clasificador devuelve una ordenación de las etiquetas conforme a su relevancia a una instancia, $\tau(\lambda)$ es la posición de la etiqueta λ en el ranking $\{1, 2, \ldots, q\}$. Se puede consultar una descripción más completa de estas métricas en [3].

III-C. Estudio de la configuración de cada algoritmo

En este análisis se pretedende determinar si hay diferencias significativas entre las diferentes configuraciones de los algoritmos objeto de estudio y, en caso de existir determinar cuál es la mejor configuración. En el caso de utilizar el puente ML, cada algoritmo es ejecutado utilizando la misma configuración y tres métodos de transformación diferentes: AT, GT y MMT. En el caso de los algoritmos específicos MIML que se han desarrollado, cada algoritmo es ejecutado utilizando la misma configuración de parámetros y tres medidas de distancias diferente: MaxH, MinH y AveH. La Tabla III muestra los resultados para la medida de Hamming loss. Debido a las limitaciones de espacio, no se han incluido las tablas para el resto de métricas de evaluación consideradas en este estudio.

Para cada métrica de evaluación, se han comparado las 3 configuraciones de cada algoritmo mediante un test de Friedman [29]. Se trata de un test no paramétrico que compara los rankings medios de los algoritmos, donde la configuración del algoritmo con el mejor valor de la métrica en un dataset recibe el ranking 1, el siguiente el ranking 2, y así sucesivamente. Finalmente, se calcula el ranking medio para cada algoritmo. Estos valores de ranking permiten conocer qué configuración obtiene los mejores resultados para una métrica considerando todos los datasets, de modo que la que obtiene el valor más cercano a 1 será la que haya obtenido mejores valores de la métrica para en un mayor número de conjuntos de datos. Los valores de ranking medios obtenidos para las diferentes configuraciones de cada algoritmo para la métrica Hamming loss se muestran en la última columna de la Tabla III.

Para determinar si alguna configuración es la mejor respecto

a las 7 métricas evaluadas, se realiza un estudio de meta-ranking [30] basado en comparar los ranking medios obtenidos por cada algoritmo para todas las métricas considerando todos los conjuntos de datos. Se puede ver un ejemplo para las configuraciones de BRkNN en la Tabla IV.

TABLA IV
META RANKING DE LAS DISTINTAS CONFIGURACIONES DE BRKNN

	Puente ML			Algoritmos MIML		
BRkNN	**AT**	**GT**	**MMT**	**AveH**	**MaxH**	**MinH**
Hloss	2,2500	2,2500	1,5000	1,5000	2,6250	1,8750
SubsetAcc	1,4375	2,3750	2,1875	0,1937	2,3125	1,7500
EBFMeasure	1,2500	2,5000	2,2500	2,1250	2,0000	1,8750
MacFMeasure	1,6250	2,3750	2,0000	1,6250	2,6250	1,7500
MicFMeasure	1,6250	2,2500	2,1250	1,7500	2,2500	2,0000
Rloss	1,3750	2,2500	2,3750	1,3750	2,7500	1,8750
EBAccuracy	1,2500	2,5000	2,2500	2,0000	2,1250	1,8750
Ranking	**1,2143**	2,7587	2,0000	**1,4286**	2,8571	1,7143

Siguiendo este procedimiento, se obtienen las Tablas V y VI que muestran los ranking medios obtenidos por cada algoritmo considerando los 8 conjuntos de datos y las 7 métricas evaluadas. La Tabla V nos permite llevar a cabo un estudio de las diferentes transformaciones del problema a ML. El mejor ranking en todos los algoritmos se obtiene con AT, lo que indica que es la transformación más beneficiosa y con menos pérdida de información. La Tabla VI nos permite llevar a cabo un estudio de la influencia de las diferentes medidas de distancia aplicadas en los algoritmos especializados para resolver el problema. En este caso no hay un claro ganador, aunque se puede concluir que AveH y MinH dan siempre mejores resultados que MaxH.

En todos los casos estudiados el $p-value < 0,05$ del test de Friedman revela que existe al menos una configuración de cada algoritmo con diferencias significativas de rendimiento a un nivel de confianza del 95 %. Por este motivo, seleccionaremos como mejor configuración aquella que ha obtenido un mejor valor de meta-ranking. En el caso de MLkNN existe un empate para AT y GT, pero al ser AT la mejor opción en el resto de algoritmos, elegiremos AT.

TABLA V
META RANKING Y TEST DE FRIEDMAN DE ALGORITMOS QUE UTILIZAN PUENTE ML ($\alpha = 0{,}05$)

	AT	**GT**	**MMT**	**Friedman gl=2**	**p-value**
BRkNN	**1,2143**	2,7587	2,0000	8,6429	0,0133
DMLkNN	**1,2143**	2,3571	2,4286	6,5000	0,0388
IBLR_ML	**1,0714**	2,8571	2,0714	11,2140	0,0037
MLkNN	**1,5714**	2,5871	**1.5714**	7,7143	0,0211
MLDGC	**1,1429**	2,1429	2,7143	8,8571	0,0119

TABLA VI
META RANKING Y TEST DE FRIEDMAN DE ALGORITMOS ESPECIALIZADOS MIML ($\alpha = 0{,}05$)

	AveH	**MaxH**	**MinH**	**Friedman gl=2**	**p-value**
BRkNN	**1,4286**	2,8571	1,7143	8,0000	0,0183
DMLkNN	**1,0000**	3,0000	2,0000	14,0000	0,0009
IBLR_ML	1,6429	2,7857	**1,5714**	6,5000	0,0388
MLkNN	1,5714	3,0000	**1,4286**	10,5710	0,0051
MLDGC	**1,2857**	3,0000	1,7143	11,1430	0,0038

III-D. Comparativa de las mejores configuraciones

Teniendo en cuenta las mejores configuraciones obtenidas en el experimento de la sección III-C, en este estudio pretendemos determinar si hay algoritmos basados en kNN que obtienen mejor rendimiento que otros en la resolución de problemas MIML. Siguiendo el mismo procedimiento que en la sección anterior, se ha realizado un estudio por métricas y se han utilizado los valores de ranking medio que consideran todas las medidas y conjuntos de datos para establecer un estudio integral. La Tabla VII muestra, para cada algoritmo, los resultados de ranking medio para cada métrica junto con el meta-ranking obtenido. El test de Friedman aplicado sobre el meta-ranking considerando todas las métricas obtiene un valor del estadístico de $23,081$ y un $p - value = 0,006018$, lo que revela que existen diferencias significativas entre los algoritmos. Los resultados del post test de Shaffer se representan de forma gráfica en la Figura 3. Se puede comprobar que el algoritmo MLDGC ofrece los mejores resultados, ya sea aplicando una transformación AT con un puente ML o bien adaptando el algoritmo a la resolución del problema utilizando la distancia AveH. El segundo algoritmo en el ranking es MLkNN, que también obtiene resultados similares tanto utilizando una transformación AT, como resolviendo el problema directamente. En este caso, utilizando la distancia MinH. Este resultado muestra que si se emplea una transformación del problema adecuada, la pérdida de información que se produce no es determinante y adquiere más relevancia la estrategia empleada por el algoritmo.

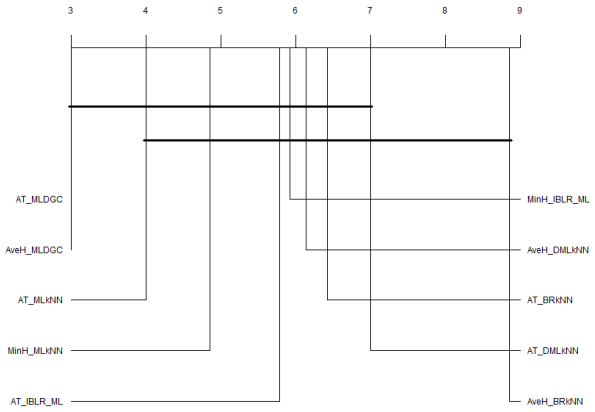


Figura 3. Distancia crítica del post test de Shaffer $\alpha = 0{,}05$

IV. CONCLUSIONES

En este trabajo se han estudiado propuestas basadas en kNN para la resolución de problemas MIML. Por un lado, se han estudiado algoritmos ML que transforman el conjunto de datos a un problema ML, y por otro se ha propuesto algoritmos especializados que resuelven directamente el problema MIML. El estudio ha determinado que la transformación utilizada en el puente ML afecta al rendimiento de los algoritmos, resultando que AT ofrece los mejores resultados en todos los algoritmos ML estudiados. Con respecto a los algoritmos MIML

TABLA VII
META RANKING DE LA MEJOR CONFIGURACIÓN DE CADA ALGORITMO

	Hloss	**SubsetAcc**	**EBFMeas**	**MacFMeas**	**MicFMeas**	**Rloss**	**EBAcc**	**Meta-ranking**
AT_BRkNN	6,8750	5,6250	6,5000	5,8125	6,0000	5,7500	6,6250	6,4286 (7)
AT_DMLkNN	5,3125	6,6875	7,3750	6,3750	7,2500	4,1250	7,3750	7,0000 (8)
AT_IBLR_ML	7,3750	5,9375	2,6250	6,7500	3,6250	7,5000	2,7500	5,7857 (4)
AT_MLkNN	3,7500	4,3125	5,6250	5,3750	5,7500	4,7500	5,6250	4,0000 (2)
AT_MLDGC	6,7500	3,5625	**3,6250**	3,3750	**3,3750**	4,3750	3,6250	3,0000 (1)
AveH_BRkNN	5,1875	7,1250	8,0000	6,6250	7,8750	6,7500	8,0000	8,8571 (9)
AveH_DMLkNN	4,7500	6,8125	7,2500	6,1875	7,1250	**3,6250**	7,2500	6,1429 (6)
MinH_IBLR_ML	7,2500	6,8750	2,8750	6,5000	3,6250	7,3750	**3,0000**	5,9286 (5)
MinH_MLkNN	4,3750	4,9375	6,0000	4,8750	6,2500	5,1250	6,0000	4,8571 (3)
AveH_MLDGC	**3,3750**	**3,1250**	5,1250	**3,1250**	4,1250	5,6250	4,7500	3,0000 (1)

especializados, la distancia de similitud también ha resultado influyente obteniendo AveH y MinH los mejores resultados. Finalmente, el estudio con las mejores configuraciones de cada algoritmo ha determinado que MLDGC y MLkNN obtienen los mejores resultados y que no existen grandes diferencias entre utilizar una transformación previa con un puente ML o una métrica de distancia específica para MIML.

AGRADECIMIENTOS

Este trabajo está financiado por el proyecto de investigación UCO-1262678 de la Junta de Andalucía y el Fondo Europeo de Desarrollo Regional.

REFERENCIAS

[1] Zhi H. Zhou and Min L. Zhang. Multi-instance multi-label learning with application to scene classification. In *NIPS*, pages 1609–1616, 2006.

[2] Thomas G Dietterich, Richard H Lathrop, and Tomás Lozano-Pérez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial intelligence*, 89(1-2):31–71, 1997.

[3] Eva Gibaja and Sebastián Ventura. A tutorial on multilabel learning. *ACM Computing Surveys (CSUR)*, 47(3):52, 2015.

[4] F. Herrera, S. Ventura., R. Bello, C. Cornelis, A. Zafra, D. Sánchez-Tarragó, and S. Vluymans. *Multiple Instance Learning. Foundations and Algorithms*. Springer, 2016.

[5] Grigorios Tsoumakas, Anastasios Dimou, Eleftherios Spyromitros-Xioufis, V. Mezaris, Ioannis Kompatsiaris, and I. Vlahavas. Correlation-based pruning of stacked binary relevance models for multi-label learning. In *Proceedings of the 1st International Workshop on Learning from Multi-Label Data*, pages 101–116, 01 2009.

[6] Lin Dong. A comparison of multi-instance learning algorithms. Master's thesis, The University of Waikato, 2006.

[7] C. Li and G. Shi. Weights optimization for multi-instance multi-label rbf neural networks using steepest descent method. *Neural Computing and Applications*, 22(7-8):1563–1569, 2013.

[8] J.S. Wu, S.J. Huang, and Z.H. Zhou. Genome-wide protein function prediction through multi-instance multi-label learning. *Transactions on Computational Biology and Bioinformatics*, 11(5):891–902, 2014.

[9] C. Tong-tong, L. Chan-juan, Z. Hai-lin, Z. Shu-sen, L. Ying, and D. Xin-miao. A multi-instance multi-label scene classification method based on multi-kernel fusion. In *Proceedings of the Conference on Intelligent Systems*, pages 782–787, 2015.

[10] M.L. Zhang. A k-nearest neighbor based multi-instance multi-label learning algorithm. In *Proceedings of the 22nd International Conference on Tools with Artificial Intelligence*, volume 2, pages 207–212, 2010.

[11] Eleftherios Spyromitros, Grigorios Tsoumakas, and Ioannis Vlahavas. An empirical study of lazy multilabel classification algorithms. In *Hellenic conference on artificial intelligence*, pages 401–406. Springer, 2008.

[12] Min-Ling Zhang and Zhi-Hua Zhou. Ml-knn: A lazy learning approach to multi-label learning. *Pattern recognition*, 40(7):2038–2048, 2007.

[13] Zoulficar Younes, Fahed Abdallah, and Thierry Denœux. Multi-label classification algorithm derived from k-nearest neighbor rule with label dependencies. In *2008 16th European Signal Processing Conference*, pages 1–5. IEEE, 2008.

[14] Weiwei Cheng and Eyke Hüllermeier. Combining instance-based learning and logistic regression for multilabel classification. *Machine Learning*, 76(2-3):211–225, 2009.

[15] Oscar Reyes, Carlos Morell, and Sebastián Ventura. Effective lazy learning algorithm based on a data gravitation model for multi-label learning. *Information Sciences*, 340:159–174, 2016.

[16] Gerald Edgar. *Measure, topology, and fractal geometry*. Springer Science & Business Media, 2007.

[17] Jun Wang and Jean-Daniel Zucker. Solving the multiple-instance problem: A lazy learning approach. In *Proceedings of the Seventeenth International Conference on Machine Learning*, ICML '00, page 1119–1126, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc.

[18] Min-Ling Zhang and Zhi-Hua Zhou. Multi-instance clustering with applications to multi-instance prediction. *Applied intelligence*, 31(1):47–68, 2009.

[19] Min-Ling Zhang and Zhi-Hua Zhou. M3miml: A maximum margin method for multi-instance multi-label learning. In *2008 Eighth IEEE International Conference on Data Mining*, pages 688–697. IEEE, 2008.

[20] Zhi-Hua Zhou and Min-Ling Zhang. Multi-instance multi-label learning with application to scene classification. In *Advances in neural information processing systems*, pages 1609–1616, 2006.

[21] Oded Maron and Aparna Lakshmi Ratan. Multiple-instance learning for natural scene classification. In *ICML*, volume 98, pages 341–349, 1998.

[22] Jian-Sheng Wu, Sheng-Jun Huang, and Zhi-Hua Zhou. Genome-wide protein function prediction through multi-instance multi-label learning. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 11(5):891–902, 2014.

[23] Forrest Briggs, Yonghong Huang, Raviv Raich, Konstantinos Eftaxias, Zhong Lei, William Cukierski, Sarah Frey Hadley, Adam Hadley, Matthew Betts, Xiaoli Z Fern, et al. The 9th annual mlsp competition: New methods for acoustic classification of multiple simultaneous bird species in a noisy environment. In *2013 IEEE international workshop on machine learning for signal processing (MLSP)*, pages 1–8. IEEE, 2013.

[24] Derek Greene and Pádraig Cunningham. A matrix factorization approach for integrating multiple data views. In *Joint European conference on machine learning and knowledge discovery in databases*, pages 423–438. Springer, 2009.

[25] Alvaro Belmonte, Amelia Zafra, Eva Gibaja, and Sebastián Ventura. Una librería para el aprendizaje multi-instancia multi-etiqueta. In *IX Simposio de Teoría y Aplicaciones de la Minería de Datos (TAMIDA 2018)*, pages 931–936, Granada, Spain, oct 2018. ISBN 978-84-09-05643-9.

[26] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18, 2009.

[27] G. Tsoumakas, E. Spyromitros-Xioufis, Jozef Vilcek, and I. Vlahavas. Mulan: A java library for multi-label learning. *Journal Machine Learning Research*, 12:2411–2414, 2011.

[28] Borja Calvo and Guzmán Santafé Rodrigo. scmamp: Statistical comparison of multiple algorithms in multiple problems. *The R Journal, Vol. 8/1, Aug. 2016*, 2016.

[29] J. Demšar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7:1–30, 2006.

[30] Alberto Cano, José María Luna, Eva Lucrecia Gibaja Galindo, and Sebastián Ventura. LAIM discretization for multi-label data. *Inf. Sci.*, 330:370–384, 2016.

Comparing BERT against traditional machine learning text classification

Santiago González-Carvajal
Universidad Politécnica de Madrid
Madrid, Spain
santiago.gonzalez-carvajal@alumnos.upm.es
Eduardo C. Garrido-Merchán
Universidad Autónoma de Madrid
Madrid, Spain
eduardo.garrido@uam.es

***Abstract*—The BERT model has arisen as a popular state-of-the-art model in recent years. It is able to cope with NLP tasks such as supervised text classification without human supervision. Its flexibility to cope with any corpus delivering great results has make this approach very popular in academia and industry. Although, other approaches have been used before successfully. We first present BERT and a review on classical NLP approaches. Then, we empirically test with a suite of different scenarios the behaviour of BERT against traditional TF-IDF vocabulary fed to ML algorithms. The purpose of this work is adding empirical evidence to support the use of BERT as a default on NLP tasks. Experiments show the superiority of BERT and its independence of features of the NLP problem such as the language of the text adding empirical evidence to use BERT as a default technique in NLP problems.**

I. Introduction

Natural Language Processing (NLP) methodologies such as text classification or summarization [9], have flourished. We can differentiate between two types of approaches to NLP problems: Firstly, linguistic approaches [2] that use different features of the text that the experts on the domain consider that are relevant. Those features could be combinations of words, or n-grams, grammatical categories, unambiguous meanings of words and much more. These features could be built manually or can be retrieved by using linguistic resources.

On the other hand, Machine Learning (ML) have analyzed annotated corpora of texts inferring which features of the text, typically in a bag of words fashion or by n-grams, are relevant for the classification automatically. Both approaches have their pros and cons, concretely, linguistic approaches have great precision but their recall is low as the context where the features are useful is not as big as the one processed by ML. However, the precision of classical NLP systems was, until recently, generally better than the one delivered by ML [4]. Nevertheless, recently, thanks to the rise of computation, ML text classification dominates.

The advantage of linguistic approaches over ML is that they do not need large amounts of data. We can find many examples of ML approaches: BERT [3], Transformers [13], etc. An issue with traditional NLP approaches is multilingualism. We can design rules for a given language, but sentence structure, and even the alphabet, may change from one language to another, resulting in the need to design new rules. Bidirectional Encoder Representations from Transformers (BERT) is a NLP model that was designed to pretrain deep bidirectional representations from unlabeled text and, after that, be fine-tuned using labeled text for different NLP tasks [3].

In this work we compare BERT model [3] with a traditional ML NLP approach that trains ML algorithms in features retrieved by the Term Frequency - Inverse Document Frequency (TF-IDF) algorithm as a representative of these traditional approaches [11]. We have carried out four different experiments about text classification. We start by presenting some related work, then, we describe the models, after that, we describe the experiments we have carried out and, finally, we present the conclusions drawn from the work.

II. Related Work

In this section, we summarize the main comparisons against advanced models such as the BERT transformer and classical natural language processing. Recently, BERT has achieved state-of-the-art results in a broad range of NLP tasks [3]. Hence, it is interesting to study how does the BERT model represent the steps of the traditional NLP pipeline in order to make a fair comparison.

An argument that defends classical ML NLP approaches is that the BERT approach need huge amounts of texts to deliver proper results. An interesting work [12] that focus on a pure empirical comparison of BERT and ULMFiT [10] w.r.t traditional NLP approaches in low-shot classification tasks where we only have 100-1000 labelled examples per class shows how BERT, representing the best of deep transfer learning, is the best performing approach, outperforming top classical ML algorithms thanks to the use of transfer learning [3]. A common critique of classical NLP practitioners is that the BERT model and ML methodologies can be fooled easily, commiting errors that may be severe in certain applications and that can be easily solved by symbolic approaches. Following this reasoning, in this work [7] the authors present the TextFooler baseline, that generates adversarial text in order to fool BERT's classification [7].

III. The BERT model and the traditional ML NLP methodology

Having reviewed related work, we will now introduce the traditional NLP approaches that we are comparing with BERT and then, the details of the BERT model.

A classical way to deal with a supervised learning NLP task is to build a bag-of-words model with the most weighted words given by the TF-IDF algorithm. Assuming there are N documents in the collection, and that term t_i occurs in n_i of these documents. Then, inverse document frequency can be computed as: $idf(t_i) = log\frac{N}{n_i}$. Actually, the original measure was an integer approximation to this formula, and the logarithm was base 2. On the other hand, given a term t_i, we denote by tf_i the frequency of the term t_i in the document under consideration. Finally, TF-IDF is defined for a given term t_i in a given document as follows: $tfidf(t_i) = tf_i \cdot idf(t_i)$.

We now explain what we consider to be the state-of-the-art technique on natural language processing. Regarding the BERT model, there are two steps in its framework: *pre-training* and *fine-tuning* [3]. During pre-training, the model is trained on unlabeled large corpus. For fine-tuning, the model is initialized with the pre-trained parameters, and all the parameters are fine-tuned using labeled data for specific tasks. BERT's model architecture is a multi-layer bidirectional Transformer encoder [3]. This kind of encoder is composed of a stack of $N = 6$ identical layers. Each of these layers has two sub-layers. The first one is a multi-head self-attention mechanism, and the second one, is a simple position-wise fully connected feed-forward network. It employs a residual connection [6] around both sub-layers, followed by a layer normalization [1]. That is, the output of each sub-layer is $LayerNorm(x + Sublayer(x))$, where $Sublayer(x)$ is the function implemented by the sub-layer [13]. In relation to, multi-head self-attention, first, we need to define scaled dot-product attention. It is define as follows: $Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V$, where Q is the matrix of queries, K is the matrix of keys, V is the matrix of values and d_k is the dimension of the Q and K matrices. Now, we can define multi-head attention as $MultiHead(Q, K, V) = Concat(head_1, ..., head_h)W^O$, where $head_i = Attention(QW_i^Q, KW_i^K, VW_i^V)$. Multi-head attention consists on projecting the queries, keys and values h times with different, learned linear projections to d_k, d_k and d_v (dimension of the values matrix), respectively. Then, on each of these projected versions of the queries, keys and values, we perform the attention function in parallel, yielding in d_v-dimensional output values. Finally, these are concatenated and projected, resulting in the final values [13]. Self-attention means that all of the keys, values and queries come from the same place.

BERT represents a single sentence or a pair of sentences (for example, the pair $\langle question, answer \rangle$) as a sequence of tokens. For a given token, its input representation is constructed by summing the corresponding token, position, and segment embeddings [3]. Pre-training is divided into: *Masked LM* and *Next Sentence Prediction (NSP)*. The first one, consists in masking some percentage of the input tokens at random, and then, predict those masked tokens. The second one consists in, given two sentences `A` and `B`, 50% of the time `B` is the actual next sentence that follows `A` (labeled as `IsNext`), and 50% of the time `B` is a random sentence from the corpus (labeled as `NotNext`) [3]. Fine-tuning is straightforward since the self-attention mechanism in the Transformer allows BERT to model many downstream tasks. For each task, we simply plug in the specific inputs and outputs into BERT and fine-tune all the parameters [3].

IV. Experiments

In order to compare BERT model with respect to the traditional ML NLP methodology, we have designed four experiments that are described throughout the section.

In these experiments, we will be using `TfidfVectorizer` from `sklearn` Python 3 module. After using TF-IDF to preprocess the text, we will be using `Predictor` from `auto_ml` module (in the third and fourth experiments), and `H2OAutoML` from `h2o` module (in the second experiment), to find the best model to fit the data. In the first experiment, we will, instead, show how much work needs to be done in order to get close to the results obtained, with no effort, using BERT model. For this purpose, we will be using many `sklearn` models and study their results in depth.

Regarding BERT's implementation, we have used the pre-trained BERT model from `ktrain` Python 3 module. This model expects the following directory structure: a directory which must contain two subdirectories: **train** and **test**. Each one of them, in turn, must contain one subdirectory per class (named after the name of the class they represent). And, finally, each class directory, must contain the `'.txt'` files (their name is irrelevant) with the texts that belong to the class they represent.

In the first experiment, we have downloaded the IMDB dataset. It contains 50000 movie reviews (25000 to train the model and 25000 to test it) to perform sentiment analysis, a popular supervised learning text classification task. We have compared the behaviour of a pre-trained default BERT model w.r.t different popular ML models such as SVC or Logistic Regression that use a vocabulary extracted from a TF-IDF model obtaining the following results:

Model	Accuracy
BERT	**0.9387**
Voting Classifier	0.9007
Logistic Regression	0.8949
Linear SVC	0.8989

TABLE I
ACCURACY RETRIEVED BY THE DIFFERENT METHODOLOGIES IN THE IMDB EXPERIMENT OVER THE VALIDATION SET.

Our second experiment deals with the RealOrNot tweets written in English. The task to solve here is binary text classification. It contains tweets classified into two different classes: Tweets about a real disaster and those that are not. We have just used the *tweet* and *class* columns. After that, we have

generated the directory structure that we need to use BERT model (using 75% data to train and 25% data to validate). The obtained results have been summarized in the following table:

Model	Accuracy	Kaggle Score
BERT	**0.8361**	**0.83640**
H2OAutoML	0.7875	0.77607

TABLE II
REALORNOT EXPERIMENT RESULTS.

Having seen that BERT has outperformed an AutoML technique and other classical ML algorithms using a vocabulary built from a traditional NLP technique such as TF-IDF in the English language, we choose to change the language to see if the BERT model also behaves well. We have downloaded the Portuguese news dataset. It contains articles from the news classified into nine different classes. We have just used the *article text* and *class* columns. We have generated the directory structure that we need to use BERT model (using 75% data to train and 25% data to validate obtaining the following results:

Model	Accuracy	Kaggle Score
BERT	**0.9093**	**0.91196**
Auto ML	0.8480	0.85047

TABLE III
PORTUGUESE NEWS EXPERIMENT RESULTS.

Our last experiment involves a completely different language, Peninsular Chinese simplified characters zh-CN, where we hypothesize that, given that the way of expressing this Language is through different symbols that are not separated by spaces BERT may not output a good result. The experiment is a sentiment analysis problem involving Chinese hotel reviews. It contains hotel reviews classified into two different classes: Positive hotel reviews and negative hotel reviews. In this experiment, we have used 85% of the data to train the model and 15% of the data to validate it. Results are given in the following table:

Model	Accuracy
BERT	**0.9381**
Predictor (auto_ml)	0.7399

TABLE IV
CHINESE HOTEL REVIEWS RESULTS.

We can observe how, independently of the language and its characteristics, BERT behaviour outperforms classical NLP approach. In this experiment, the importance of transfer learning has become apparent, since the dataset was pretty small.

V. CONCLUSIONS AND FURTHER WORK

In this work we have introduced the BERT model and the classical NLP strategy where a ML model is trained using the features retrieved with TF-IDF and hypothesize about the behaviour of BERT w.r.t to tackle NLP tasks. We have introduced four different NLP scenarios where we have shown how BERT has outperformed the traditional NLP approach. Critically, implementing BERT has turned out to be far less complicated than implementing traditional methods. It is also noteworthy the importance of transfer learning. We have been able to obtain this results thanks to pre-training, as it become apparent in the last experiment. We are nevertheless aware of the limitations of the BERT model. Although it seems a good default for NLP tasks, it can be improved. Hence, we would like to apply Bayesian optimization for BERT to enable classification of messages for robots [8] [5] showing consciousness correlated behaviours.

ACKNOWLEDGMENTS

Authors gratefully acknowledge the use of the facilities of Centro de Computacion Cientifica (CCC) at Universidad Autónoma de Madrid. The authors also acknowledge financial support from Spanish Plan Nacional I+D+i, grants TIN2016-76406-P and from PID2019-106827GB-I00 /AEI /10.13039/501100011033.

REFERENCES

[1] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.
[2] Erik Cambria and Bebo White. Jumping nlp curves: A review of natural language processing research. *IEEE Computational intelligence magazine*, 9(2):48–57, 2014.
[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.
[4] Eduardo C Garrido and Jesús Cardenosa Lera. Expansión supervisada de léxicos polarizados adaptable al contexto.
[5] Eduardo C Garrido-Merchán, Martin Molina, and Francisco M Mendoza. An artificial consciousness model and its relations with philosophy of mind. *arXiv preprint arXiv:2011.14475*, 2020.
[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
[7] Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. Is bert really robust? natural language attack on text classification and entailment. *arXiv preprint arXiv:1907.11932*, 2019.
[8] Eduardo C Garrido Merchán and Martín Molina. A machine consciousness architecture based on deep learning and gaussian processes. *arXiv preprint arXiv:2002.00509*, 2020.
[9] Cristina Puente, José Angel Olivas, E Garrido, and R Seisdedos. Creating a natural language summary from a compressed causal graph. In *2013 joint ifsa world congress and nafips annual meeting (ifsa/nafips)*, pages 513–518. IEEE, 2013.
[10] Kristian Rother and Achim Rettberg. Ulmfit at germeval-2018: A deep neural language model for the classification of hate speech in german tweets. 2018.
[11] Bruno Trstenjak, Sasa Mikac, and Dzenana Donko. Knn with tf-idf based framework for text categorization. *Procedia Engineering*, 69:1356–1364, 2014.
[12] Peter Usherwood and Steven Smit. Low-shot classification: A comparison of classical and deep transfer machine learning approaches. *arXiv preprint arXiv:1907.07543*, 2019.
[13] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

Aprendizaje de clasificadores binarios justos mediante Optimización Multiobjetivo

Jorge Casillas Brranquero
Departamento de Ciencias de la
Computación e Inteligencia Artificial
Universidad de Granada
Granada, España
0000-0002-5887-3977

David Villar Martos
Universidad de Granada
Granada, España
Daalma7@correo.ugr.es

***Resumen*—La justicia en el Aprendizaje Automático es un área que está cobrando gran relevancia. Los resultados de algunos algoritmos de aprendizaje automático se han demostrado sesgados en contra de ciertos grupos demográficos históricamente desfavorecidos debido a diversos factores. Utilizando métodos de clasificación justos podemos disminuir estos efectos, aunque existen múltiples definiciones de justicia mutuamente incompatibles que se pueden aplicar.**

El objetivo es proponer un algoritmo que proporcione clasificadores binarios justos respecto a un atributo sensible binario. Puesto que existen definiciones de justicia incompatibles, utilizaremos una aproximación de optimización Many Objectives, buscando soluciones precisas, justas respecto a 2 criterios contradictorios e interpretables. El espacio de decisión vendrá dado por los hiperparámetros de clasificación base, en nuestro caso Árboles de decisión.

Dicho algoritmo ha sido probado con 4 conjuntos de datos bien estudiados en el campo de la justicia en el aprendizaje automático, dando buenos resultados con soluciones variadas permitiendo seleccionar un clasificador con unas características concretas en función de nuestras necesidades.

***Palabras clave*—Justicia en el aprendizaje automático, Árboles de decisión, Optimización Many Objectives.**

I. Introducción

La justicia en el Aprendizaje Automático está cobrando relevancia en los últimos años como campo de investigación, aunque los problemas relacionados con la misma estaban documentados desde la concepción de la disciplina, más concretamente en el ámbito de la estadística. Los humanos estamos sesgados frente a ciertos aspectos, y dichos sesgos se reflejan en la construcción de las bases de datos con las que entrenar modelos, aprendiendo éstos dichos sesgos.

Los problemas relacionados con la justicia se han vuelto más acuciantes durante estos últimos años [1], [2] debido a una variedad de factores, como la conciencia social o distintos casos mediáticos que han despertado el interés por parte de la comunidad [3], [4]. Estos hechos se han detectado en una gran variedad de campos, tales como los derechos humanos, privacidad, empleo, o la salud [3]–[6].

Existen varias formas de implementar definiciones de justicia en nuestros problemas, realizaremos una clasificación no mutuamente excluyente en tres clases [7]: preprocesado, en fase de entrenamiento y postprocesado. La primera de éstas aboga por modificar la base de datos del problema para eliminar posibles focos de sesgo en los datos. Las técnicas en fase de entrenamiento modifican el propio funcionamiento del algoritmo de forma que tengan en cuenta y aplique definiciones de justicia en el proceso de aprendizaje. Normalmente involucra un proceso de optimización con restricciones [8]. Por último, las técnicas de postprocesado alteran las funciones una vez aprendidas para que éstas sean más justas. El método desarrollado pertenece a la clase de algoritmos que incorporan técnicas en fase de entrenamiento.

La definición matemática de qué consideraremos justo [9] es un tema no trivial y necesario para poder desarrollar metodologías. En la actualidad podemos encontrar un amplio espectro de definiciones matemáticas [10] basadas en diferentes conceptos utilizados en el ámbito jurídico como tratamiento desigual o impacto desigual. Las principales familias de definiciones de justicia son la justicia de grupo, individual y contrafactual. Las definiciones basadas en las dos últimas se valen en gran medida de estructuras y funciones altamente dependientes del contexto, mientras que las definiciones de la familia de justicia de grupo tienen una definición estrictamente probabilística, lo que hace que se puedan aplicar a una gran cantidad de ámbitos. Sin embargo, algunas de estas definiciones son incompatibles entre sí [11] y su uso estará condicionado a la perspectiva con la que tratar el contexto. Se ha demostrado que esta compensación ocurre también entre calidad en las predicciones y los criterios de justicia [12]. Al poder cuantificar el grado de justicia y precisión de un algoritmo en base a una definición, ello insta a encontrar clasificadores que optimicen dichas métricas llegando hasta el compromiso insalvable que hay entre ellas. Métodos que consideran definiciones de justicia se han visto capaces de encontrar soluciones con una calidad equiparable o mejor que otros ampliamente utilizados, garantizando además una decisión menos discriminatoria [13]. Pese a ello, al utilizar árboles de decisión, puede que esta optimización conduzca a clasificadores muy complejos y difícilmente interpretables, limitando altamente su aplicabilidad, por lo que será interesante considerar un objetivo relacionado con dicha interpretabilidad.

El problema de optimización a resolver se interpretará por tanto como un problema Many Objectives. Estos problemas se definen como problemas de optimización multiobjetivo con más de dos objetivos. Sean dos conjuntos $\Omega \subset \mathbb{R}^n, \Theta \subset \mathbb{R}^m : m > 2$, representando a los espacios de decisión y objetivo respectivamente. Sea una función $(f_1, ...f_m) = f : \Omega \to \Theta$, donde cada f_i será una función objetivo a minimizar. Buscaremos encontrar las soluciones que cumplan

$$\min_{\omega \in \Omega}(f_1(\omega), f_2(\omega), ..., f_m(\omega)) \tag{1}$$

Puesto que los objetivos son contradictorios, no habrá ningún valor en el espacio de decisión que pueda minimizar todos los objetivos simultáneamente. Es por esto que surge el concepto de dominancia de Pareto: sean $\psi, \omega \in \Omega$ soluciones a un problema de minimización multiobjetivo. Diremos que ψ domina a ω en el sentido de Pareto, notado por $\psi \prec \omega$ sí y sólo sí se cumplen las siguientes propiedades:

$$\begin{gathered} f_i(\psi) \leq f_i(\omega), \forall i \in \{1, 2, ..., m\} \\ \exists j \in \{1, 2, ..., m\} : f_j(\psi) < f_j(\omega) \end{gathered} \tag{2}$$

De esta forma buscaremos nuestro objetivo será encontrar un conjunto de soluciones que no estén dominadas en el sentido de Pareto por ninguna otra, las cuales se conocen como conjunto Pareto optimal. Sus imágenes por la función objetivo se conoce como Frente Pareto. Para realizar este procedimiento se utilizarán Algoritmos Evolutivos de Optimización Many Objectives (MaOEAs), los cuales se diferencian con respecto a los tradicionales en que tienen técnicas para paliar los efectos negativos que supone el aumento de la dimensionalidad de cara a la optimización, y tienen una eficacia probada y contrastada en problemas reales [14]

II. Objetivos, definiciones de justicia

Nuestro problema lo enfocaremos como un problema de optimización de cuatro objetivos, teniendo uno de precisión, dos objetivos contradictorios de la familia de justicia de grupo, y un objetivo de interpretabilidad del árbol de decisión que generemos.

Notaremos como A al atributo sensible, Y como la variable a predecir, y p será la predicción hecha por nuestro clasificador. Tanto A como Y serán variables binarias, y por lo tanto p también lo será, pudiendo tomar cada una de ellas los valores 0 y 1

Para evaluar la precisión en la predicción de nuestro árbol de decisión usaremos el criterio de la media geométrica, combinando las medidas de la tasa de verdaderos positivos ($P[p = 1|Y = 1]$) y la tasa de verdaderos negativos ($P[p = 0|Y = 0]$). Utilizamos esta medida porque tiene en cuenta ambas tasas, penalizando los casos en que alguna de ellas sea muy baja. Puesto que el problema es de minimización, utilizaremos

$$\text{Error} = 1 - \sqrt{P[p = 1|Y = 1] \cdot P[p = 0|Y = 0]}$$

Vamos a utilizar dos medidas de justicia pertenecientes a la familia de justicia de grupo, más concretamente a las subfamilias de probabilidades igualadas y de paridad en el ratio predictivo, las cuales son contradictorias. [11]

La subfamilia de probabilidades igualadas entiende que el valor de la predicción hecha condicionado al verdadero valor de su etiqueta debe ser independiente al valor que tome el atributo sensible A, y por lo tanto $P[p|Y, A = 0] = P[p|Y, A = 1]$ para alguna combinación de valores de p e Y [15]. Nosotros vamos a considerar exclusivamente la tasa de falsos positivos, $P[p = 1|Y = 0]$, y mediremos el grado de justicia respecto a este valor, calculando la diferencia de la tasa de las tasas de falsos positivos condicionada a ambos valores de A.

$$\text{FPR}_{\text{diff}} = |P[p = 0|Y = 1, A = 0] - P[p = 0|Y = 1, A = 1]|$$

Se alcanzará un mínimo cuando se cumpla la definición de probabilidades igualadas para falsos positivos. Es evidente que en caso de darse, también se cumpliría para verdaderos positivos. Esta medida puede suponer problemas de impacto desigual en la población.

La segunda definición de justicia está comprendida en la familia de paridad en el ratio predictivo, donde se considera ahora que el valor real del atributo a predecir condicionado al valor de la predicción debe ser independiente del atributo sensible, y por lo tanto $P[Y|p, A = 0] = P[Y|p, A = 1]$. En ese caso, al igual que antes consideramos el ratio de falsos positivos, vamos a utilizar el valor predictivo positivo $P[Y = 1|p = 1]$, y estableceremos la medida de justicia como la diferencia entre los valores predictivos positivos condicionados a que el atributo sensible tome cada valor:

$$\text{PPV}_{\text{diff}} = |P[Y = 1|p = 1, A = 0] - P[Y = 1|p = 1, A = 1]|$$

Se cumpliría la definición de justicia cuando ambos valores sean iguales y la diferencia valga 0.

En este caso podría haber problemas considerando de impacto desigual también. Utilizando las mismas condiciones del ejemplo anterior, la solución propuesta también cumpliría el criterio de justicia de PPV_{diff}.

El intentar optimizar estos objetivos anteriormente descritos puede suponer generar clasificadores muy complejos y poco interpretables, lo que reduciría la utilidad de un modelo puesto que no se podría justificar su decisión y su uso como sistemas de soporte a la decisión estaría muy limitado. Por tanto, queremos que los modelos sean interpretables, lo que en este caso equivale a generar árboles simples. Para ello utilizaremos como criterio el número de hojas del árbol. Con este valor podemos ver la cantidad de posibles caminos de decisión que existen en el árbol, y por lo tanto la interpretación total de los procesos de decisión que lleva a cabo el árbol dependerán de menos características y será más fácil interpretarlo.

III. Espacio de decisión, hiperparámetros

La implementación de los árboles de decisión que utilizaremos será la de la clase DecisionTreeClassifier de la biblioteca scikit-learn. Utilizaremos como espacio de decisión los siguientes 5 hiperparámetros pertinentes:

- criterion: Representa la función que medirá la calidad de la división de un nodo intermedio. El árbol de decisión

genera nuevos nodos a partir de los que ya hay en niveles inferiores conforme este valor se va haciendo más pequeño. Existen dos posibles valores, siendo Gini y Entropía.
- max_depth: Máxima profundidad que puede tener el árbol. Los árboles más profundos tienden a ser más complejos que aquellos que no lo son. Los posibles valores que puede tomar son $\{n \in \mathbb{N} : n \geq 3\}$, además de un valor que representa que no hay restricción en este aspecto, lo notaremos por ∞.
- min_samples_split: Mínimo número de muestras que necesitaremos que caigan sobre un nodo para poder dividirlo. Conforme mayor es este número, más sencillo será el que un nodo se divida. Los posibles valores que consideraremos serán $\{n \in \mathbb{N} : 2 \leq n \leq 40\}$.
- max_leaf_nodes: Máximo número de hojas que el árbol puede tener. Conforme mayor es, más complejo será el árbol puesto que mayor número de caminos distintos se podrán tomar hasta llegar a una clasificación final. Los posibles valores son $\{n \in \mathbb{N} : n \geq 2\}$, además de un valor que representa que no hay restricción en este aspecto, lo notaremos por ∞.
- class_weight: Este parámetro da un peso para cada clase, lo cual es útil para conjuntos de datos desbalanceados. Toma valores en el intervalo $[0, 1]$, representando un valor de $\frac{1}{2}$ que ambas clases tendrán el mismo peso. Los posibles valores que usaremos son $\{\frac{n}{10} : n \in \mathbb{N}, 1 \leq n \leq 9\}$

IV. Algoritmos empleados

Para la resolución de nuestro problema de optimización Many Objectives utilizaremos tres algoritmos distintos, NSGA-II, SMS-EMOA y GrEA.

IV-A. Método basado en NSGA-II

Como base se usará el algoritmo NSGA-II, algoritmo clásico de referencia en optimización multiobjetivo, y el cual ha mostrado ser efectivo en optimizaciones para precisión y justicia [13]. La implementación que realizaremos será muy similar a la realizada en dichos estudios y utilizaremos los mismos parámetros puesto que se evidencia que tienen una buena eficacia. Además de la optimización para 4 objetivos (NSGA-II$_4$) probraremos a optimizar para 2 objetivos sólamente (NSGA-II$_2$) siendo el objetivo de error y FPR$_{\text{diff}}$, como se hizo en **??** pero calculando también los otros dos objetivos, para comprobar cómo evolucionan las soluciones encontradas al pasar a optimización de 4 objetivos.

NSGA-II no es un algoritmo evolutivo explícitamente diseñado para Many Objectives, por lo que es especialmente sensible a efectos adversos debido a la alta dimensionalidad del espacio objetivo, tales como la resistencia a la dominancia ó la promoción activa de diversidad [16], con lo que su desempeño puede verse deteriorado. Por tanto hemos seleccionado dos MaOEAs del estado del arte basados en conceptos diferentes que han mostrado tener un buen comportamiento en entornos Many Objectives con el objetivo de ofrecer una gama variada de soluciones y algoritmos.

IV-B. Método basado en SMS-EMOA

El segundo algoritmo que utilizaremos será SMS-EMOA ($\mathcal{S}$ metric Selection-EMOA) [17], perteneciente a la clase de MaOEAs basados en en medidas indicadoras, en este caso el hipervolumen. El algoritmo se vale de la S métrica, también conocida como hipervolumen para dirigir la búsqueda como criterio de reemplazamiento.

El algortimo genera un único individuo en cada generación, y tras añadirlo a la población, se elimina de la misma al individuo que menos contribuya al hipervolumen total de toda la población. Se los operadores originalmente propuestos de cruce SBX y mutación polinómica. Para la selección se usará un esquema de torneo binario donde se calculará un ranking de no dominancia y se preferirá individuos con mejor posición en dicho ranking. En caso de que este criterio falle, se usará la crowding distance, al igual que se utiliza en NSGA-II.

Para el cálculo del hipervolumen debemos de establecer el peor punto del espacio objetivo. Para hacerlo de una forma independiente de los objetivos usados y para no sesgar a priori el cálculo del hipervolumen a favor ni en contra de ningún objetivo, normalizaremos los objetivos que no tomen naturalmente valores en $[0, 1]$ para que sí lo hagan. En este caso sólo sucede con el número de hojas. De esta forma el peor punto del espacio objetivo será el punto con un 1 en todas sus coordenadas. En general el cálculo del hipervolumen es costoso computacionalmente si el número de individuos u objetivos es muy alto probablemente haya otros metros preferibles como el siguiente propuesto.

IV-C. Método basado en GrEA

El algoritmo GrEA (Grid based Evolutionary Algorithm) [18] es un MaOEA perteneciente a la clase de dominancia relajada, y dentro de la misma, a los MaOEAs con dominancia basada en valor [16].

Lo que caracteriza a este algoritmo es introducir un entorno de rejilla o grid adaptativo en el espacio objetivo en función de la población, y definir operadores en base a medidas sobre dicho grid. En particular, para la selección utilizaremos los conceptos de dominancia de Pareto, dominancia de grid y grid crowding distance. Para el reemplazo generacional utilizaremos la valoración de grid, grid crowding distance y distancia de grid al punto coordenada. El operador de cruce es el mismo que NSGA-II.

V. Experimentación

Para nuestra experimentación utilizaremos 4 conjuntos de datos reales estudiados y de gran relevancia en el campo de la justicia en el aprendizaje automático:

- Adult: Información de ciudadanos estadounidenses en 1994. Posee 32561 instancias y 14 atributos. El atributo protegido es race. El problema de clasificación consistirá en saber si un individuo gana más de $50k al año o menos.
- German: Información financiera sobre individuos. Posee 1000 instancias y 20 atributos. El atributo protegido con-

siderado es age. Se busca conocer el riesgo de devolución del crédito concedido a una persona entre alto y bajo.

- ProPublica: Información sobre el funcionamiento del algoritmo COMPAS, sobre individuos del condado de Broward, Florida, durante los años 2013 y 2014. Hay 7214 instancias, con 52 atributos. Los atributos se han reducido a los utilizados en [19], siendo: sex, age, age_cat, race, juv_fel_count, juv_misd_count, juv_other_count, priors_count, c_charge_degree, c_charge_desc, decile_score y score_text. El atributo protegido es race. Se busca predecir si el individuo en cuestión reincidirá (volviendo a ser arrestado) de nuevo en los próximos 2 años.
- ProPublica violent: Mismo conteto que en Propublica, pero esta vez queremos predecir si el individuo volverá a ser arrestado en los próximos 2 años por un crimen violento. Contiene 4743 individuos, considerando las mismas 12 características y atributo protegido que en el caso anterior.

Emplearemos nuestros algoritmos para dar un conjunto de soluciones para estos problemas, para los cuales usaremos un total de 300 generaciones y un tamaño de población de 150 individuos. Los parámetros establecidos para los métodos que así los necesiten serán los mismos que los empleados en [13]. Realizaremos 10 ejecuciones distintas con cada algoritmo, donde cambiaremos la semilla aleatoria que afecta a la partición de datos entre entrenamiento y test, manteniendo la proporción de 75 % para entrenamiento y 25 % para test, así como otros procesos aleatorios del algoritmo.

Para tener un conjunto de referencia con el que comparar la calidad de los conjuntos de soluciones, calcularemos los individuos no dominados de entre todas las ejecuciones y algoritmos, es decir, los mejores individuos encontrados, que supondrán la mejor aproximación posible al frente Pareto real.

Además, utilizaremos una serie de medidas de calidad para evaluar la calidad de los conjuntos de soluciones. Notaremos como PF a nuestra aproximación del frente Pareto, y A será el conjunto de soluciones del que queremos calcular las medidas.

- Hipervolumen: Definido como $\int_{(0,1)^m} \mathbb{1}_{D_A}(z)dz$ donde $D_A = \{z = (z_1, ..., z_m) \in (0,1)^m : \exists a \in A : a \prec z\}$ y $\mathbb{1}_{D_A}$ siendo la función característica del conjuntoD_A. Cuanto mayor, mejor será el conjunto de soluciones desde una perspectiva general respecto a la convergencia, distribución y uniformidad de las soluciones.
- Distancia generacional (GD): Definida como $\frac{(\sum_{i=0}^{|A|} d_i^2)^{1/2}}{|A|}$, donde d_i se calcula para la solución $x_i \in A$ como $\min\{||f(x_i) - f(y)||_2 : y \in PF\}$. Cuanto menor, mayor convergencia del conjunto A respecto al conjunto PF habrá en media.
- Distancia generacional invertida: Definida como $\frac{(\sum_{i=0}^{|PF|} d_i^2)^{1/2}}{|PF|}$, donde d_i se calcula para cada $x_i \in PF$ como $\min\{||f(x_i) - f(y)||_2 : y \in A\}$. Es similar a GD, pero la perspectiva es distinta, esta medida es más recomendable para analizar la representatividad de las soluciones respecto a PF de forma que si hay una región poco explorada esta medida aumentará.
- Proporción: $\frac{|B|}{|PF|}$ donde $B = \{x \in A \cap PF\}$
- Ratio de error: $\frac{|B|}{|A|}$ donde $B = \{x \in A \wedge x \notin PF\}$
- Espaciado: Definido como $\sqrt{\frac{1}{|A|-1}\sum_{i=1}^{|A|}(\overline{d} - d_i)^2}$, donde $d_i = \min_{x_j \in A, x_j \neq x_i} \{||f(x_i) - f(x_j)||_1\}$, y $\overline{d} = \frac{\sum_{i=0}^{|A|} d_i}{|A|}$. Cuanto mayor sea, habrá más distancia entre cada solución y su solución más próxima en media.
- Dispersión máxima: Definida como $||z||_2$ donde $z_\alpha = \max_{x_i, x_j \in A} |f_\alpha(x_i) - f_\alpha(x_j)|$. Cuanto mayor, habrá valores más extremos en las coordenadas del espacio objetivo.
- Dispersion general de frente Pareto: Definida como $\prod_{i=1}^{m} \frac{|\max_{x \in A} f_i(x) - min_{x \in A} f_i(x)|}{|f_i(P_N) - f_i(P_I)|}$, donde P_N es la a aproximación del punto nadir, y P_I es una aproximación del punto ideal [20], las cuales calcularemos de la siguiente forma: $P_{I_\alpha} = \min_{x \in PF} f_\alpha(x)$, and $P_{N_\alpha} = \max_{x \in PF} f_\alpha(x)$. Cuanto más cercana a 1, mejor se mantiene la distribución entre las mejores y peores soluciones en cada objetivo en la población A con respecto a PF.

VI. Resultados

Vamos a analizar cómo el valor de cada objetivo cambia cuando pasamos de una optimización de 2 a 4 objetivos. Tomaremos la media de las soluciones medianas de cada ejecución de NSGA-II$_2$ como referencia, respecto al error en validación. Posteriormente, haremos una media ponderada de los 2 individuos de cada ejecución del resto de algortimos justo por encima y los 2 justo por debajo del valor medio encontrado en error de validación en función de cómo de lejos estén en error en validación de dicho valor encontrado, siendo menos significativos cuanto más lejos estén. Haciendo este análisis podemos ver cómo se comportan los algoritmos bajo un mismo error. Los resultados los podemos ver en la tabla I

Como podemos ver, las soluciones que se han encontrado por nuestros algoritmos de 4 objetivos se comportan mejor en los objetivos en los que nuestro algoritmo de dos 2 no estaba preparado para optimizar, pero por contra se comporta peor en $\text{FPR}_{\text{diff, v}}$. Todos ellos se comportan mejor en PPV_{val}. Ocurre en general lo mismo con el número de hojas. GrEA es capaz de obtener la mitad respecto al número de hojas de NSGA-II$_2$ de forma consistente. Este comportamiento es esperable puesto que el algoritmo de 2 objetivos tiende a mejorar mucho en error y $\text{FPR}_{\text{diff, v}}$ sin tener en cuenta los otros 2 objetivos, mientras los algoritmos de optimización de 4 objetivos generan soluciones mucho más equilibradas, y estos nuevos objetivos juegan en contra de $\text{FPR}_{\text{diff, v}}$.

Realizaremos ahora un análisis del cubrimiento de las soluciones Pareto optimales respecto a todas las ejecuciones de un algoritmo para cada conjunto de datos. El cubrimiento es una medida de calidad que sirve para comparar dos conjuntos de soluciones A y B. Se calcula como $C(A,B) = \frac{\{b \in B : \exists a \in A : a \prec b\}}{|B|}$. Cuanto mayor, mejores soluciones habrá en A respecto de las que hay en B. Los resultados de cubrimiento se mostrarán en una matriz, en cuya diagonal pondremos la

Cuadro I: Objective values of weighted average of individuals near the median on NSGA-II$_2$ on validation error

		2 Objectives		4 Objectives				
		$Error_v$	$FPR_{diff,v}$	$PPV_{diff,v}$	N.Leaves	$Error_t$	$FPR_{diff,t}$	$PPV_{diff,t}$
Adult	NSGA-II$_2$	0.2362	0.0169	0.1163	256.8000	0.2390	0.0346	0.0715
	NSGA-II$_4$	0.2362	0.0413	0.0741	278.5704	0.2366	0.0410	0.0770
	SMSEMOA	0.2362	0.0329	0.0604	210.8909	0.2374	0.0333	0.0722
	GrEA	0.2362	0.0330	0.0391	137.9508	0.2364	0.0303	0.0605
German	NSGA-II$_2$	0.3155	0.0202	0.1631	38.3000	0.3498	0.1041	0.0972
	NSGA-II$_4$	0.3177	0.0891	0.1240	30.7936	0.3387	0.1215	0.0854
	SMSEMOA	0.3172	0.0933	0.0941	34.3890	0.3436	0.1101	0.1035
	GrEA	0.3190	0.0967	0.1021	20.4307	0.3516	0.0872	0.1312
ProPub	NSGA-II$_2$	0.3708	0.0290	0.0922	258.8000	0.3822	0.0533	0.0820
	NSGA-II$_4$	0.3708	0.0869	0.0520	158.9334	0.3722	0.0894	0.0602
	SMSEMOA	0.3708	0.0504	0.0704	113.0509	0.3758	0.0742	0.0709
	GrEA	0.3708	0.0612	0.0577	107.6457	0.3765	0.0777	0.0676
ProP. V.	NSGA-II$_2$	0.3645	0.0235	0.1223	102.2000	0.3798	0.0474	0.0991
	NSGA-II$_4$	0.3645	0.0781	0.0779	92.8691	0.3816	0.0716	0.0957
	SMSEMOA	0.3645	0.0603	0.0703	99.5176	0.3763	0.0642	0.0898
	GrEA	0.3645	0.0686	0.0700	57.5332	0.3770	0.0696	0.0915

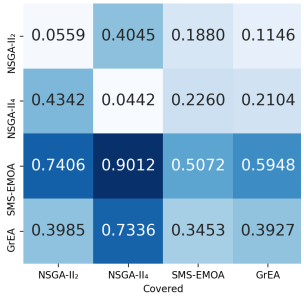
proporción de individuos Pareto optimales aportados por cada algoritmo considerando todos los individuos Pareto optimales de entre todos los algoritmos y ejecuciones juntos, es decir, los mejores individuos encontrados para dicho conjunto de datos, a los que llamaremos conjunto Pareto Optimal general (PO general) De esta forma podemos comprobar en proporción cuánto mejor son las soluciones de un algoritmo respecto a otro, además de cuántas soluciones no dominadas por ningún otro algoritmo y por tanto exclusivas es cada algoritmo capaz de proponer. Además haremos dos rankings, uno con la cantidad de veces en las que $C(A, B) > C(B, A)$ para todas las parejas de soluciones en todos los conjuntos de datos, y otro en el que se calcule la media de la proporción de orígen de las mejores soluciones para todos los conjuntos de datos. Los resultados se pueden ver en la figura 1

Ahora podemos ver claramente que SMS-EMOA y GrEA han tenido en general un mejor comportamiento que ambas variantes de NSGA-II. Estos algoritmos ganan en ambos rankings por bastante diferencia. En general representan la mayor parte de soluciones PO generales y cubren mayor proporción de soluciones de lo que son cubiertas. El conjunto de datos de German se escapa de la tendencia general, debido a que es un conjunto con un considerable menor número de instancias, donde el algoritmo NSGA-II$_4$ se desenvuelve mejor. Se puede ver cómo en general para el caso de optimización Many Objectives con los criterios seleccionados es preferible el uso de MaOEAs.

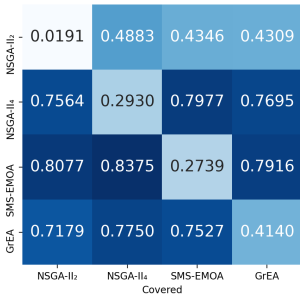
En la tabla **??** podemos ver las medidas de calidad citadas de calidad sobre los individuos Pareto optimales para los algoritmos de optimización con 4 objetivos. El conjunto PO con el que comparar se calcula sólo usando esos algoritmos. Las medidas que se pueden calcular sobre el mismo también se muestran. Las soluciones de SMS-EMOA están generalmente más convergidas hacia el Frente Pareto, seguidas por las de GrEA, y la distancia generacional invertida de SMS-EMOA también es la más baja, excepto en German, lo que significa que las soluciones han convergido mucho y en general representan bien al conjunto PO general. El ratio de error es consistente con los datos de proporción, ya que cuando uno aumenta, el otro disminuye, de lo que podemos inferir que el número de soluciones no dominadas encontradas por cada algoritmo ha sido relativamente similar. No hay clara evidencia de que las medidas de distribuciones estén relacionadas con la calidad de las soluciones.

Cuadro II: Many-objective quality measures obtained in each problem

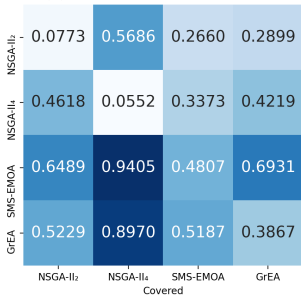
Data	Quality Measure	NSGA-II	SMS-EMOA	GrEA	General PF
Adult	Hypervolume	0.8190	0.8249	0.8164	0.8249
	GD	0.0068	0.0017	0.0023	-
	Inverted GD	0.0040	0.0035	0.0032	-
	Proportion	0.1114	0.4764	0.4122	1.0000
	Error ratio	0.6432	0.1159	0.2507	-
	Spacing	0.0229	0.0212	0.0162	0.0179
	Maximum spread	1.4379	1.1036	1.2532	1.4840
	Overall PF spread	0.9302	0.1386	0.5340	-
German	Hypervolume	0.7162	0.7189	0.7133	0.7190
	GD	0.0067	0.0086	0.0107	-
	Inverted GD	0.0082	0.0052	0.0099	-
	Proportion	0.3256	0.3101	0.3643	1.0000
	Error ratio	0.0968	0.1563	0.3493	-
	Spacing	0.0505	0.0443	0.0457	0.0404
	Maximum spread	0.8996	1.1768	0.9737	1.2161
	Overall PF spread	0.5217	0.7261	0.4982	-
ProPublica	Hypervolume	0.6683	0.6766	0.6708	0.6765
	GD	0.0076	0.0034	0.0034	-
	Inverted GD	0.0057	0.0037	0.0052	-
	Proportion	0.1689	0.4374	0.3937	1.0000
	Error ratio	0.5643	0.1843	0.2791	-
	Spacing	0.0257	0.0285	0.0194	0.0218
	Maximum spread	1.0338	1.2271	0.8307	1.2578
	Overall PF spread	0.6219	0.8109	0.2359	-
ProPublica V.	Hypervolume	0.6850	0.6903	0.6833	0.6903
	GD	0.0103	0.0041	0.0051	-
	Inverted GD	0.0074	0.0063	0.0074	-
	Proportion	0.1703	0.4444	0.3853	1.0000
	Error ratio	0.6678	0.1276	0.4061	-
	Spacing	0.0927	0.0288	0.0487	0.0542
	Maximum spread	1.8360	1.0903	1.2151	1.8374
	Overall PF spread	0.9974	0.0503	0.0590	-



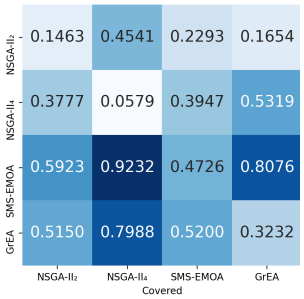
(a) Cubrimientos Adult



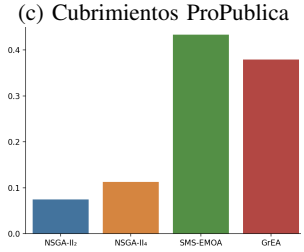
(b) Cubrimientos German



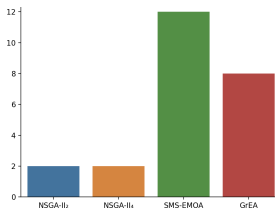
(c) Cubrimientos ProPublica



(d) Cubrimientos ProPublica V.



(e) Media de la distribución del conjunto PO general



(f) Ranking de dominancia

Figura 1: Matrices de cubrimiento junto con las distribuciones del conjunto PO general para todos los conjuntos de datos y algoritmos, así como rankings asociados

VII. Conclusiones

Se ha introducido un nuevo método que genera una amplia gama de modelos de clasificación optimizados utilizando una serie de objetivos contradictorios, incluyendo definiciones de justicia. El método utiliza hiperparámetros de árboles de decisión como espacio de decisión, por lo que se entrena una gran cantidad de árboles con una amplia gama de distribuciones de los objetivos que buscamos optimizar, entre los que se incluyen objetivos de justicia. Puesto que nos enfrentamos a un problema de optimización Many objectives, optamos por utilizar MaOEAs pertenecientes al estado del arte siendo SMS-EMOA y GrEA. Mostramos cómo es posible obtener buenos resultados de justicia en las familias de probabilidades igualadas y paridad en el ratio predictivo, manteniendo unos buenos niveles de precisión e interpretabilidad, y por lo tanto existe espacio para mejorar en términos de justicia sin verse perjudicada la precisión.

Posteriores estudios podrían involucrar utilizar otros tipos de clasificadores y criterios. Añadir más criterios puede incrementar la diferencia en la calidad de las soluciones de NSGA-II con respecto a los MaOEAs empleados. Se podría utilizar un clasificador que incluyera de forma explícita justicia en su criterio de aprendizaje, de forma que hubiera un parámetro que controlara cómo de relevante sería este criterio en el proceso, lo que añadiría un parámetro relevante al espacio de decisión

Referencias

[1] A. D. Selbst, D. Boyd, S. A. Friedler, S. Venkatasubramanian, and J. Vertesi, "Fairness and abstraction in sociotechnical systems," in *Proceedings of the Conference on Fairness, Accountability, and Transparency*, 2019, pp. 59–68.

[2] R. Binns, M. Van Kleek, M. Veale, U. Lyngs, J. Zhao, and N. Shadbolt, "'it's reducing a human being to a percentage' perceptions of justice in algorithmic decisions," in *Proceedings of the 2018 Chi conference on human factors in computing systems*, 2018, pp. 1–14.

[3] S.C. of the United States, "Ricci v. DeStefano," Tech. Rep. U.S. 557, 2009.

[4] J. Angwin, J. Larson, S. Mattu, and L. Kirchner, "Machine bias," *ProPublica, May*, vol. 23, p. 2016, 2016.

[5] T. Bolukbasi, K.-W. Chang, J. Y. Zou, V. Saligrama, and A. T. Kalai, "Man is to computer programmer as woman is to homemaker? debiasing word embeddings," in *Advances in neural information processing systems*, 2016, pp. 4349–4357.

[6] M. Zehlike, F. Bonchi, C. Castillo, S. Hajian, M. Megahed, and R. Baeza-Yates, "Fa* ir: A fair top-k ranking algorithm," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, 2017, pp. 1569–1578.

[7] S. A. Friedler, C. Scheidegger, S. Venkatasubramanian, S. Choudhary, E. P. Hamilton, and D. Roth, "A comparative study of fairness-enhancing interventions in machine learning," in *Proceedings of the conference on fairness, accountability, and transparency*, 2019, pp. 329–338.

[8] M. B. Zafar, I. Valera, M. Gomez-Rodriguez, and K. P. Gummadi, "Fairness constraints: A flexible approach for fair classification." *J. Mach. Learn. Res.*, vol. 20, no. 75, pp. 1–42, 2019.

[9] N. Mehrabi, F. Morstatter, N. Saxena, K. Lerman, and A. Galstyan, "A survey on bias and fairness in machine learning," 2019.

[10] P. Gajane and M. Pechenizkiy, "On formalizing fairness in prediction with machine learning," 2018.

[11] E. del Barrio, P. Gordaliza, and J.-M. Loubes, "Review of mathematical frameworks for fairness in machine learning," 2020.

[12] A. K. Menon and R. C. Williamson, "The cost of fairness in binary classification," in *Conference on Fairness, Accountability and Transparency*, 2018, pp. 107–118.

[13] A. Valdivia, J. Sánchez-Monedero, and J. Casillas, "How fair can we go in machine learning? Assessing the boundaries of accuracy and fairness," *International Journal of Intelligent Systems*, vol. 36, no. 4, pp. 1619–1643, 2021.

[14] K. Narukawa and T. Rodemann, "Examining the performance of evolutionary many-objective optimization algorithms on a real-world application," in *2012 Sixth International Conference on Genetic and Evolutionary Computing*, 2012, pp. 316–319.

[15] M. Hardt, E. Price, and N. Srebro, "Equality of opportunity in supervised learning," 2016.

[16] B. Li, J. Li, K. Tang, and X. Yao, "Many-objective evolutionary algorithms: A survey," *ACM Computing Surveys (CSUR)*, vol. 48, no. 1, pp. 1–35, 2015.

[17] M. Emmerich, N. Beume, and B. Naujoks, "An emo algorithm using the hypervolume measure as selection criterion," in *International Conference on Evolutionary Multi-Criterion Optimization*. Springer, 2005, pp. 62–76.

[18] S. Yang, M. Li, X. Liu, and J. Zheng, "A grid-based evolutionary algorithm for many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 17, no. 5, pp. 721–736, 2013.

[19] S. A. Friedler, C. Scheidegger, S. Venkatasubramanian, S. Choudhary, E. P. Hamilton, and D. Roth, "A comparative study of fairness-enhancing interventions in machine learning," 2018.

[20] C. Audet, J. Bigeon, D. Cartier, S. Le Digabel, and L. Salomon, "Performance indicators in multiobjective optimization," *European journal of operational research*, 2020.

Análisis predictivo efectivo en centros de datos: una solución basada en inteligencia artificial

David García Retuerta
BISITE Research Group
Universidad de Salamanca
Calle Espejo 2, 37007,
Salamanca, España
0000-0003-2351-068X

Roberto Casado-Vara
BISITE Research Group
Universidad de Salamanca
Calle Espejo 2, 37007,
Salamanca, España
rober@usal.es

Juan M. Corchado
BISITE Research Group
Universidad de Salamanca
Calle Espejo 2, 37007,
Salamanca, España
corchado@usal.es

***Abstract*—El mantenimiento predictivo a tenido un papel fundamental en garantizar el buen funcionamiento de distintas industrias durante los últimos años. Las técnicas de potenciales fallos o defectos de producción en etapas tempranas tienen la capacidad de evitar errores más graves. En este caso, analizamos el funcionamiento de un gran centro de datos con distintos sistemas informáticos interconectados. El sistema sufre de fallos periódicos de naturaleza aparentemente caótica que lo fuerzan a reiniciarse, quedando inoperativo durante largos periodos de tiempo. El objetivo de este trabajo es diseñar un método capaz de realizar avisos sobre potenciales errores antes de que ocurran, y dar cierta información sobre los mismos. Diversas técnicas matemáticas y de inteligencia artificial han sido estudiadas (por ejemplo minería de procesos, series temporales y reglas de asociación). Los mejores resultados han sido obtenidos por medio de la detección de valores atípicos en el número de logs y la aparición de ciertas palabras clave.**

***Index Terms*—Mantenimiento Predictivo, Centro de Datos, Inteligencia Artificial, Outliers, NLP**

I. Introducción

La industria 4.0 anticipa el principio del fin de las aplicaciones centralizadas tradicionales para el control de la producción. Su visión de los ecosistemas de las fábricas inteligentes es descentralizada por naturaleza. Respondiendo a las demandas del cliente con productos a medida, estas plantas están repletas de alta tecnología como impresoras 3D, dispositivos IoT, Big Data, Computación en la nube, Edge Computing, dispositivos móviles, etc. Los sistemas productivos del futuro tienen que ser construidos y diseñados teniendo en cuenta este cambio de paradigma [2].

Una de las partes fundamentales de cualquier industria del futuro es el mantenimiento predictivo de sus productos. Se trata de la última política de mantenimiento que ha sido adoptada por numerosos sectores, especialmente aquellos con funciones esenciales (por ejemplo plantas eléctricas, sistemas de transporte, comunicaciones, servicios de emergencia, etc). En esencia, el mantenimiento predictivo se encarga de predecir fallos o problemas en sistemas en deterioro con el fin de optimizar los esfuerzos de mantenimiento. Gracias a ello el mantenimiento se vuelve más proactivo, efectivo y eficiente [12].

Una de las técnicas que más pueden impulsar el mantenimiento predictivo es la inteligencia artificial y, en concreto, el machine learning. En un entorno de industria 4.0, los datos generados por las redes de sensores requieren de un procesad basado en análisis de datos [6]. El machine learning ofrecen a las distintas organizaciones la posibilidad de tener un algoritmo capaz de detectar patrones y formular predicciones, así como de mejorarse a sí mismo automáticamente basado en su experiencia [7].

Gracias a la inteligencia artificial ha habido grandes avances en forma de optimizaciones en el campo manufacturero. Esto ha resultado en la creación de las denominadas “Smart Factories” [4]. Estas fabricas inteligentes son capaces de integrar sistemas de detección de errores en tiempo real, analizar las transformaciones de un productor a lo largo de toda la cadena de montaje o de calcular la calidad de un determinado producto instantáneamente.

Todos estos métodos también son de vital importancia para cualquier smart city o smart territory en desarrollo. Las ciudades inteligentes suelen dividir sus áreas de actuación en los siguientes verticales [8]: Smart governance, Smart economy, Smart mobility, Smart environment, Smart people y Smart living. En concreto, el dominio de Smart economy abarca la innovación, la productividad y el espíritu emprendedor. Todos estos aspectos pueden resultar gratamente beneficiados por la implantación de técnicas de inteligencia artificial o de mantenimiento predictivo. El uso de dispositivos, máquinas y sistemas inteligentes crea una tendencia hacia la digitalización del territorio, estimulando la innovación y permitiendo a las compañías tomar decisiones mejor informadas [5].

Revolucionarias técnicas han sido probadas con éxito en los últimos años para el desarrollo de soluciones prácticas útiles para grandes compañías. Por ejemplo, las técnicas de

Esta investigación ha sido financiada por el proyecto “Intelligent and sustainable mobility supported by multi-agent systems and edge computing (InEDGE-Mobility): Towards Sustainable Intelligent Mobility: Blockchain-based framework for IoT Security”, Referencia: RTI2018-095390-B-C32, financiado por el Ministerio Español de Ciencia, Innovación y Universidades (MCIU), la Agencia Estatal de Investigación (AEI) y el Fondo Europeo de Desarrollo Regional (FEDER).

Esta investigación ha sido parcialmente financiada por el proyecto “Computación cuántica, virtualización de red, edge computing y registro distribuido para la inteligencia artificial del futuro”, Referencia: CCTT3/20/SA/0001, financiado por el Instituto para la competitividad empresarial de Castilla y León, y el fondo europeo para el desarrollo regional (FEDER).

minería de procesos han sido examinadas de forma preliminar en el ámbito de las empresas globales de tecnologías de la información [9], el análisis de outliers ha sido implementado con éxito en diversas aplicaciones industriales [11], y las series temporales han sido satisfactoriamente utilizadas para tratar datos temporales en problemas de mantenimiento predictivo en empresas de semiconductores [3].

Este artículo está dividido de la siguiente manera: la sección II describe el sistema creado y los algoritmos que el mismo implementa, así como los resultados obtenidos. Tras ello, la sección III presenta otros algoritmos probados para realizar un mantenimiento predictivo, pero cuyo rendimiento ha resultado inferior. Finalmente, la sección IV presenta las conclusiones obtenidas en el presente estudio y las futuras lineas de investigación.

II. Propuesta

El sistema desarrollado constada de dos módulos independientes para la detección de posibles fallos o errores. Ambos funcionan en paralelo y se complementan entre sí, cada uno lanzando una clase de alertas distinto.

Los errores pueden deberse a un error recurrente que se va magnificando a través del sistema conforme pasa el tiempo, ciertos comandos que funcionan como desencadenantes de un error en casada o simplemente errores humanos con consecuencias casi inmediatas. Los errores a corto y medio plazo han sido modelados por medio de los métodos descritos a continuación.

A. Detección de valores atípicos

En el caso estudiado, analizando las incidencias acontecidas se llega a la siguiente conclusión: el número de logs registra un aumento inusual en las horas o minutos previos a las incidencias. Con el objetivo en mente de ser capaz de detectar dicho aumento, se ha desarrollado un sistema basado en outliers.

Matemáticamente un outlier se define como aquel valor que se encuentra fuera de 1,5 veces el rango intercuartílico, centrado en la mediana.

En este caso, se crean las listas a partir de los logs correspondientes a las cinco horas previas al momento actual, agrupando los mensajes en clusters de cinco minutos. Un outlier se define como un valor atípico encontrado en una lista de los mismos. El número de clusters que es considerado como outlier en este contexto es lo que se compara con el valor umbral.

De cara a compensar la naturaleza de los sistemas con los que se trata, se han definido dos hiper-parámetros para cada sistema:

- Número mínimo de logs: Cuando una lista no alcanza el número mínimo de elementos, sus alarmas son ignoradas. En los casos analizados su valor ha oscilado entre 30 y 40.
- Umbral: porcentaje mínimo de logs que han de ser considerados outliers para que se emita una alerta. En los casos considerados su valor se ha fijado en un 15%.

Por lo tanto, el sistema funciona de la siguiente manera: se establece como input la lista de logs de las últimas cinco horas (agrupados en clusters de cinco minutos).

Tras ello, el sistema calcula cuántos de estos clusters pueden ser considerados valores atípicos. Un valor es considerado atípico si se cumple una de las siguientes condiciones:

$$\begin{cases} valor < Q1 - 1.5 * IQR & \text{o bien} \\ valor > Q3 + 1.5 * IQR \end{cases}$$

siendo IQR el rango intercuartílico y Q1, Q3 el primer y tercer cuartil (respectivamente).

En caso de que el número de outliers obtenidos sea superior al hiper-parámetro umbral, una alerta es emitida. Ante varias alertas sucesivas, un operario deberá tomar las medidas oportunas para evitar una potencial incidencia.

Por otro lado, si el número total de logs en las últimas cinco horas es inferior al hiper-parámetro número mínimo de logs, ninguna alerta será emitida.

Esta técnica ha sido aplicada con éxito en varios sistema. En estos casos se han recuperado los logs previos a las incidencias ocurridas desde septiembre de 2018 y han sido procesados en todos los casos. A modo de ejemplo, en el siguiente gráfico se muestra el número de logs en cada cluster (azul), la incidencia (rojo) y las alertas emitidas (amarillo).

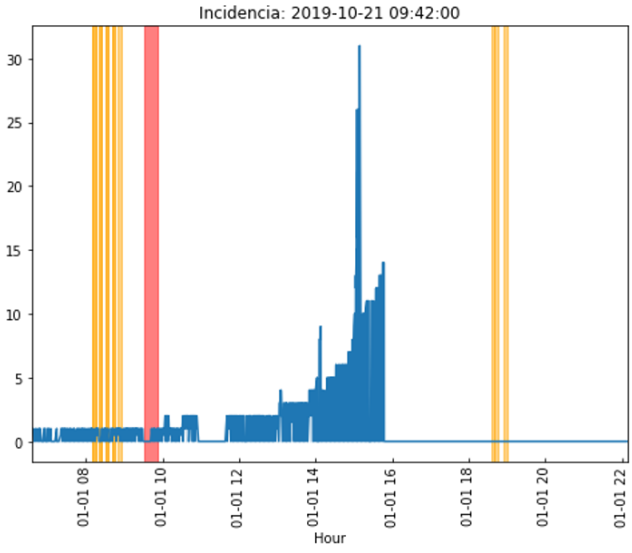


Fig. 1. Ejemplo de visualización de análisis basado en valores atípicos.

Cabe destacar que en este caso los falsos positivos serían descartados, pues no se llega al número mínimo de logs establecido como hiper-parámetro.

La intensidad de las alertas enviadas al operario viene dada por el número de alertas generadas en las últimas horas.

En general un gran número de incidencias son predichas satisfactoriamente por este método, demostrando gran versatilidad para adaptarse a diversas situaciones. Sin embargo, también se detectan algunas incidencias en potencia que nunca llegan a materializarse (falsos positivos) y se dan casos de

alertas cuya antelación a la incidencia es demasiado baja como para tomar medidas que la eviten.

B. Detección de palabras clave

La técnica de NLP (Procesamiento del Lenguaje Natural, por sus siglas en inglés) desarrollada se basa en el algoritmo de extracción de palabras clave RAKE (Rapid Automatic Keyword Extraction) [10].

Este algoritmo hace uso de delimitadores y stopwords para dividir el texto. Tras ello, se crea una matriz de co-ocurrencias compuesta por las key-frases encontradas entre delimitadores o stopwords. Calculando el grado de cada palabra dentro de la matriz, se obtiene una serie de puntuaciones para cada grupo de palabras. De este modo, los diez grupos de palabras con la puntuación más alta son considerados las keywords/key-frases.

Sería posible ampliar el rango de uso de esta técnica en un futuro, añadiendo nuevas expresiones al diccionario creado, contrastando manualmente cuales de ellas sólo se encuentran antes de las incidencias.

El sistema funciona utilizando la técnica de procesamiento de lenguaje natural RAKE, con la cual se extraen las palabras más importantes de los logs previos a las incidencias ocurridas. Otras técnicas de NLP (por ejemplo, N-gramas) también han sido probadas y desechadas, pues han mostrado un peor rendimiento. Tras analizar las palabras clave obtenidas con RAKE caso a caso, se ha podido descubrir que ciertas combinaciones de palabras son un indicador unívoco de la existencia de una incidencia en los próximos minutos u horas. Un diccionario de dichas expresiones se ha creado al efecto. El análisis se ha verificado y contrastado frente a una gran cantidad de datos para verificar que no haya falsas ocurrencias, resultando satisfactorio.

También se ha estudiado el cotejamiento de dichas palabras con el resto del dataset para descartar las que se den en situaciones de funcionamiento normal, evitando así posibles falsos positivos.

Este sistema, por lo tanto, funciona emitiendo alertas cada vez que una de las combinaciones de palabras es encontrada en los logs, siendo la búsqueda efectuada en tiempo real. Las alertas están diseñadas para ser mostradas al operario, indicando el sistema del que proceden y la alta probabilidad de que se produzca una incidencia. A modo de ejemplo, Fig. 2 muestra la nube de palabras correspondiente a uno de los sistemas analizados utilizando la técnica RAKE.

III. Métodos

Diversos métodos de análisis adicionales han sido analizados debido a sus prometedoras capacidades en el análisis de datos y su posible buena adaptación al caso propuesto. A continuación se describen los más interesantes de ellos.

A. Minería de procesos

El process mining consiste en una técnica de administración de procesos, que permite analizarlos de acuerdo a un registro de eventos. A través de esta actividad se desea extraer conocimiento a partir de los registros de evento de



Fig. 2. Wordcloud de keywords de uno de los sistemas analizados.

los procesos almacenados por los sistemas. Este conocimiento implica lograr realizar la traza de los procesos en estudio, incluyendo información de los actores que lo realizan o los tiempos involucrados, entre otras características [14].

El objetivo de esta técnica, pues, es comparar el flujo de los procesos en una situación normal contra el flujo previo a una incidencia. En la Fig. 3 se puede apreciar un ejemplo real del mismo. Esta técnica ha sido probada en diversos sistemas.

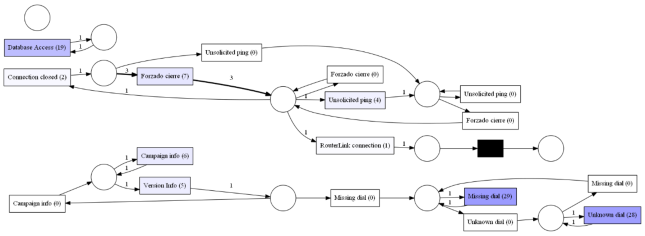


Fig. 3. Ejemplo process mining en Altitude

No ha sido posible apreciar ningún cambio significativo en los procesos antes de las incidencias, ni en sus relaciones, ni en el tiempo medio de las mismas. Por ello la técnica no ha sido seleccionada para el sistema puesto en producción.

B. Series temporales

El objetivo de una serie temporal reside en estudiar los cambios en una variable con respeto al tiempo (descripción), y en predecir sus valores futuros (predicción). Por lo tanto, el análisis de series temporales presenta un conjunto de técnicas estadísticas que permiten extraer las regularidades que se observan en el comportamiento pasado de la variable, para tratar de predecir el comportamiento futuro [13].

En este caso se ha buscado modelar el comportamiento del número de logs agrupados en clusters de cinco minutos, para predecir un aumento inusual de su número (indicativo de incidencia). Esta técnica ha sido probada en un sistema contra las variables de los dos sistemas cuyo funcionamiento está más íntimamente relacionado al primero. En la Figura 4 puede apreciarse un ejemplo de los resultados obtenidos.

Como puede observarse las predicciones no modelan correctamente la naturaleza del sistema. Esto probablemente se deba a la naturaleza caótica de los sistemas considerados, lo cual

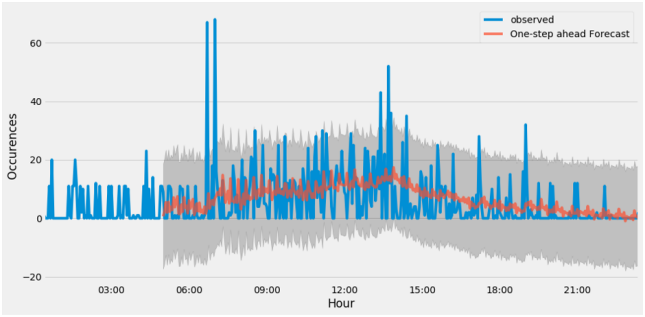


Fig. 4. Ejemplo series temportales aplicadas a Altitude. Linea de casos observados (azul), línea de casos predichos (naranja) y el área de incertidumbre (sombrado gris).

no los hace fácilmente modelables con técnicas generalistas. Como se muestra en el gráfico, la predicción no es capaz de realizar los cambios bruscos que sí se dan en el caso real; llegando estas variaciones incluso a salirse del intervalo máximo de incertidumbre de la predicción.

C. Reglas de asociación

Las reglas de asociación son una técnica de aprendizaje automático que permite obtener automáticamente qué elementos se dan interrelacionados con frecuencia en los logs. Se pueden emplear tanto a nivel de mensaje (i.e., entre los campos de una misma línea) como a nivel de líneas contiguas (posiblemente definiendo un criterio de ordenación como alternativa a la ocurrencia de estas en el log) [1].

Una regla de asociación sencilla es del tipo p->q, significando que cuando ocurre un antecedente p suele ocurrir también un consecuente q.

La validez de la regla se puede medir con la llamada confianza (confidence), un número cuyo valor oscila entre 0 y 1 representando la fracción de veces que se cumple la regla. Esto es, la fracción de casos en que los que aparece p y en que también aparece q. Otra magnitud que se puede considerar es el soporte (support), que indica la cantidad de veces que aparecen ambos en el log. Una última magnitud de interés es el lift, que mide si la observación puede ser casualidad y que en realidad los elementos implicados en la regla sean independientes. En tal caso el lift tomaría valores cercanos a 1. Si están muy alejados (mayores o menores), apunta a que exista una relación causal entre ellos.

Esta técnica ha sido probada en todos los sistemas y se han encontrado las siguientes reglas de asociación interesantes de cara a predecir incidencias:

- easy_init.exe -> Incidencia
- tsapi-avaya-definity-aes-3.1/tsapi-avaya-definity-aes-3.1.exe -> Incidencia
- pure-ftpd -> Incidencia

Estas reglas parecen haber encontrado una correlación entre la aparición de los tres procesos anteriores y la aparición de incidencias. Sin embargo, este algoritmo no ha sido seleccionado debido a una mezcla de varios factores, entre los que destacan: el reducido número de reglas encontradas y el elevado grado de computación necesario para implementar la técnica en el caso real. Debido al ser una técnica muy específica y que detecta muy pocas incidencias, no aporta suficiente valor como para ser utilizado habitualmente. A continuación puede encontrarse un ejemplo de las reglas de asociación obtenidas para uno de los sistemas considerados en concreto (Fig. 5).

```
{'antecedent': frozenset({'sleeping'}),
 'confidence': 0.9999548288011564,
 'consequent': frozenset({'TASK MANAGER Micr
 'lift': 5.137518925643853,
 'support': 0.17303994371922146}
==========================================
{'antecedent': frozenset({'30'}),
 'confidence': 0.9993804213135069,
 'consequent': frozenset({'pandora'}),
 'lift': 65.16347466801068,
 'support': 0.012608457750332213}
==========================================
{'antecedent': frozenset({'wwwHermes01'}),
 'confidence': 0.9989155973251401,
 'consequent': frozenset({'apache:'}),
 'lift': 18.878899743803395,
 'support': 0.04320331431251466}
```

Fig. 5. Ejemplo reglas de asociación en uno de los sistemas.

IV. Conclusiones

Este trabajo propone un algoritmo híbrido de inteligencia artificial para realizar un mantenimiento predictivo en centros de datos. El algoritmo utiliza logs de los servidores como entrada de datos y ha sido probado con los datos recolectados en un escenario real. El objetivo principal del sistema creado es obtener un método capaz de generar alertas que avisen a los administradores sobre posibles incidencias próximas, todo ello de forma automática.

Las incidencias son detectadas mediante el modelado de valores atípicos del número de logs y mediante la generación e inspección de palabras claves. Gracias a ellos los administradores del sistema tienen la posibilidad de evitar varios tipos de incidencias en potencia o de, al menos, mitigar las incidencias que han detectado. Además, diversos métodos alternativos han sido probados y descartados pues su rendimiento es sensiblemente inferior al de los algoritmos seleccionados

El nivel de versatilidad del sistema basado en outliers es muy elevado, siendo capaz de detectar gran variedad de incidencias. En concreto, muestra un buen funcionamiento en las incidencias con diagnóstico “Instancia offline”. Se trata de un sistema capaz de tener un buen rendimiento ante situaciones nuevas y nunca antes vividas.

El nivel de fiabilidad de las alertas obtenidas a partir del estudio palabras clave es muy elevado -del 100% en algunos casos-, pero únicamente son capaces de detectar un reducido número de incidencias. Las alertas generadas por este sistema han de ser muy tenidas en cuenta pues se trata de escenarios

conocidos, en los cuales se esperan encontrar pocas falsas alarmas.

Las futuras lineas de investigación en este aspecto incluyen mejorar la extracción de palabras clave, ampliando el número de posibles incidencias que este método puede detectar; reducir el número de falsos positivos del predictor basado en outliers mediante otras técnicas matemáticas; y el testeo de otras técnicas de inteligencia artificial que puedan adaptarse bien al problema considerado.

REFERENCES

[1] Agrawal, R., Mannila, H., Srikant, R., Toivonen, H., Verkamo, A.I., et al.: Fast discovery of association rules. Advances in knowledge discovery and data mining **12**(1), 307–328 (1996)

[2] Bogoviz, A.V.: Industry 4.0 as a new vector of growth and development of knowledge economy. In: Industry 4.0: Industrial Revolution of the 21st Century, pp. 85–91. Springer (2019)

[3] Chen, A., Blue, J.: Recipe-independent indicator for tool health diagnosis and predictive maintenance. IEEE Transactions on Semiconductor Manufacturing **22**(4), 522–535 (2009)

[4] Chen, B., Wan, J., Shu, L., Li, P., Mukherjee, M., Yin, B.: Smart factory of industry 4.0: Key technologies, application case, and challenges. Ieee Access **6**, 6505–6519 (2017)

[5] Corchado, J.M., Chamoso, P., Hernández, G., Gutierrez, A.S.R., Camacho, A.R., González-Briones, A., Pinto-Santos, F., Goyenechea, E., Garcia-Retuerta, D., Alonso-Miguel, M., et al.: Deepint. net: A rapid deployment platform for smart territories. Sensors **21**(1), 236 (2021)

[6] Diez-Olivan, A., Del Ser, J., Galar, D., Sierra, B.: Data fusion and machine learning for industrial prognosis: Trends and perspectives towards industry 4.0. Information Fusion **50**, 92–111 (2019)

[7] Garcia-Retuerta, D., Casado-Vara, R., Calvo-Rolle, J.L., Quintián, H., Prieto, J.: Deep learning for house categorisation, a proposal towards automation in land registry. In: International Conference on Hybrid Artificial Intelligence Systems. pp. 698–705. Springer (2020)

[8] Garcia-Retuerta, D., Chamoso, P., Hernández, G., Guzmán, A.S.R., Yigitcanlar, T., Corchado, J.M.: An efficient management platform for developing smart cities: Solution for real-time and future crowd detection. Electronics **10**(7), 765 (2021)

[9] Gupta, M., Serebrenik, A., Jalote, P.: Improving software maintenance using process mining and predictive analytics. In: 2017 IEEE International Conference on Software Maintenance and Evolution (ICSME). pp. 681–686. IEEE (2017)

[10] Rose, S., Engel, D., Cramer, N., Cowley, W.: Automatic keyword extraction from individual documents. Text mining: applications and theory **1**, 1–20 (2010)

[11] Rousopoulou, V., Nizamis, A., Giugliano, L., Haigh, P., Martins, L., Ioannidis, D., Tzovaras, D.: Data analytics towards predictive maintenance for industrial ovens. In: International Conference on Advanced Information Systems Engineering. pp. 83–94. Springer (2019)

[12] Selcuk, S.: Predictive maintenance, its implementation and latest trends. Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture **231**(9), 1670–1679 (2017)

[13] Shih, S.Y., Sun, F.K., Lee, H.y.: Temporal pattern attention for multivariate time series forecasting. Machine Learning **108**(8), 1421–1441 (2019)

[14] Van Der Aalst, W.: Process mining. Communications of the ACM **55**(8), 76–83 (2012)

Uso de técnicas de Inteligencia Artificial en el ámbito de la Atención Temprana

Ignacio Sierra, Norberto Díaz-Díaz, Carlos Barranco
Computer Science Division- Intelligent Data Analysis (DATAi)
Universidad Pablo de Olavide
Sevilla, España
{isierui, ndiaz, cbarranco}@upo.es

Rocío Carrasco-Villalón
Centro Atención Temprana
Hospital San Juan de Dios
Sevilla, España
rocio.carrasco@sjd.es

***Resumen*—El uso de la Inteligencia Artificial para la ayuda a la toma de decisiones es una aproximación extendida en la actualidad. Esta aplicación se encuentra en diferentes ámbitos, siendo el médico uno de los más disruptivos. Sin embargo, existen pocos o nulas aplicaciones en Atención Temprana que permitan la ayuda al diagnóstico y la asignación automática en los procesos de terapia a los niños que son atendidos en estos centros.**

El objetivo de este trabajo es realizar una primera aproximación al problema y realizar una prueba de concepto real que permita demostrar que este tipo de sistemas pueden ser útiles en Atención Temprana donde el diagnóstico y posterior tratamiento deben ser determinados por un equipo multidisciplinar.

Para medir la calidad del uso de este tipo de tecnologías se usaran diferentes técnicas de aprendizaje automático sobre un conjunto de datos real suministrado por el Hospital San Juan de Dios. Este estudio nos permitirá analizar el comportamiento de estas técnicas frente al diagnóstico tradicional. Para realizar esta comparación se contará con un punto vista cualificado en el ámbito del diagnostico de niños.

***Index Terms*—Inteligencia Artificial, Atención temprana, ayuda a la toma de decisiones, diagnóstico automático.**

I. Introducción

La Atención Temprana proporciona cuidados y servicios especiales a los niños y jóvenes con diversidad funcional o retrasos en el desarrollo. Engloba una serie de medidas, tratamientos y programas dirigidos a los niños y niñas de entre 0 y 6 años con trastornos en el desarrollo o riesgos de padecerlos. El objetivo es minimizar los efectos negativos que estos trastornos puedan ocasionar en el crecimiento y evolución de los pequeños y reduciendo las limitaciones. Asimismo proporciona herramientas a los padres y otros adultos importantes en la vida de estos niños de cara a que puedan disponer de mecanismos y conocimientos para que estos niños puedan tener experiencia y oportunidades que les ayuden a desarrollar habilidades que les serán de utilidad en su día a día.

La importancia de intervenciones tempranas ha sido reconocida por organismos internacionales como la UNESCO [22] o la *European Agency for Development in Special Needs Education* [6]. Estos estudios ponen de manifiesto la importancia y beneficios de una elección de un tratamiento acertado [14], así como la complejidad. Como en cualquier tratamiento médico, el diagnóstico preciso es imprescindible para realizar un tratamiento correcto, sin embargo, este proceso es más complejo, si cabe, debido al tipo de paciente al que va dirigido (pediatría) y al carácter multidisciplinar que posee el área de atención temprana. Disciplinas como desordenes cognitivos, desordenes del lenguaje, desordenes físicos, desordenes emocionales y desordenes de comportamiento, forman parte de este área donde el profesional debe hacer el diagnostico basándose principalmente en su experiencia subjetiva, observación y otros estudios de desarrollo del sujeto por encima de las pruebas patológicas como ocurre en otras disciplinas. En este sentido, consideramos que el uso de la Inteligencia Artificial para ayudar al profesional en el proceso de diagnostico y clasificación del paciente, puede jugar un papel muy relevante.

Centrándonos en la aplicación de Inteligencia Artificial (IA) en el ámbito de la medicina, podemos encontrar un gran volumen de su aplicación en áreas como el diagnostico, descubrimientos de nuevos medicamentos, cirugía o medicina personalizada [12]. Sin embargo, aunque como pone de manifiesto el estudio de Reddy Rajula [18], existen muchos estudios realizados en las diferentes áreas de la medicina pero no hay tantos que se ocupen del área de la pediatría.

La inteligencia artificial (IA) es capaz detectar algunas enfermedades con niveles de precisión similares a los de los profesionales sanitarios. La ventaja principal de la utilización de IA es la capacidad de examinar miles de expedientes médicos para identificar patrones de enfermedad. Esto ofrece un enorme potencial para mejorar la precisión y la velocidad del diagnóstico. En pocos casos estos algoritmos han podido superar a equipos médicos experimentados, pero si han probado su utilidad como mecanismo de ayuda al diagnostico y apoyo a equipos menos experimentados. Este tipo de sistemas son de gran utilidad en áreas complejas o multidisciplinares como pueden ser el diagnostico de desordenes psiquiátricos [11], [15]. Trabajos como el desarrollado por Davenport y Kalakota [5], ponen de manifiesto el gran potencial que la aplicación de la IA puede tener en el ámbito de la medicina.

Esta aplicación exitosa de la IA en diferentes ámbitos de la medicina, nos hace pensar que también pueda ser de utilidad en Atención Temprana, área en donde no existen aproximación de este tipo. El desarrollo de estas soluciones permitiría reducir

Esta investigación ha sido financiada por el V Plan Propio de Investiación y Trasnferencia de la Universidad Pablo de Olavide (PPI1901) y por la Junta de Andalucía, bajo el Plan de Investigación de Andalucía (TIC-239)

el tiempo necesario para el diagnostico y ayudaría al doctor a asignar adecuadamente el proceso a seguir por parte del paciente. Nótese que una clasificación inicial adecuada acelera el proceso de mejora de los pacientes.

La idea de este trabajo es realizar una primer análisis de la hipótesis de que la IA pueda ayudar a diagnosticar un paciente de atención temprana, recomendando uno u otro tratamiento a partir de la información recopilada por el especialista y el médico de atención primaria. Con tal fin, se analizarán el comportamiento de diferentes técnicas de aprendizaje automático en un entorno de pruebas real, donde se comparará el resultado obtenido con el diagnostico real de un especialista. Este articulo describe la metodología seguida, los desafíos encontrados así como los resultados alcanzados.

II. Metodología

Para el desarrollo de esta investigación se ha elegido la metodología CRISP-DM (CRoss-Industry Standard Process for Data Mining) [2]. Esta metodología, propone seis etapas diferenes: comprensión de objetivos, comprensión de los datos, preparación de los datos, modelado, evaluación y despliegue.

Seguidamente, se detallará la aproximación propuesta atendiendo a cada una de las fases previamente enunciadas.

II-A. Comprensión de objetivos

En primer lugar, nos centramos en entender los requerimientos desde un punto de vista de resultados objetivos de la investigación a realizar, transformando el problema que queremos resolver en un problema de inteligencia artificial. Para ello debemos definir muy claramente los objetivos a alcanzar.

Uno de los principales desafíos en la atención temprana es la necesidad de un trabajo interdisciplinar de varios especialistas, o una persona con una preparación en distintas disciplinas, para realizar un diagnóstico adecuado. Cuando el diagnostico se lleva a cabo por equipos multidisciplinares es frecuente que se produzcan solapamientos entre los distintos especialistas, debido a que en estos estadios de desarrollo del paciente, es común que el diagnostico no esté claro y evolucione a lo largo del tiempo. Adicionalmente, e incluso cuando las responsabilidades y áreas de los distintos especialistas está clara, la interpretación de cada uno de ellos puede no coincidir. Para lidiar con estas posibles discrepancias podemos encontrar trabajos como el de Rocio et al. [19] [Rocio Carrasco Villalon, Coordinadora del Centro Atención Temprana en Hospital San Juan de Dios Sevilla] en el que se especifica un proceso a seguir para simplificar el diagnóstico. Este enfoque, adicionalmente, evita que los niños y sus familiares reciban informaciones dispares de diferentes especialistas, se les pueda explicar de forma concisa el itinerario que debe seguir el paciente y, por ende, aliviar la presión en la que se ven envueltos en estas situaciones. Con este fin, los diagnósticos y tratamientos se organizan en procesos, limitando así la cantidad de éstos. Concretamente se especifican cuatro procesos diferentes:

1. ***Cognitivos***, también conocidos como trastornos neurocognitivos (NCDs), esta es una categoría donde se engloban desordenes de salud mental que fundamentalmente afectan a las habilidades cognitivas con pueden ser el aprendizaje, la memoria, la percepción y la resolución de problemas.
2. Desordenes de ***comunicación y lenguaje*** que es un desorden comunicativo que dificulta el aprendizaje y el uso de cualquier tipo de lenguaje hablado, escrito, lenguaje de signos...
3. ***Sensoriomotor***: son aquellos que tienen síntomas físicos como el dolor, la falta de fuerza, falta de movilidad etc.
4. ***Socio-comunicativo*** frecuentemente asociados a trastornos del comportamiento.

De esta manera, la asignación de un proceso u otro al paciente se convierte en parte esencial de su tratamiento. Contar con un sistema inteligente de ayuda a la toma de decisión podría aliviar la presión del médico y hacer más certera la asignación realizada.

El objetivo de este trabajo es realizar una prueba de concepto en la que se analice la viabilidad de desarrollo un sistema de ayuda a la toma de decisión basado en el uso de inteligencia artificial que clasifique a un paciente en un proceso de tratamiento concreto. Es de reseñar que esta clasificación puede resultar especialmente compleja, requiriendo años de experiencia a los expertos, debido a que el paciente se encuentra en pleno desarrollo cognitivo y motriz, con lo que es común que cambie de de proceso a lo largo de su tratamiento.

II-B. Comprensión de datos

La comprensión de los datos comienza con una recopilación inicial de éstos, nuestra familiarización con el conjunto de partida para la identificación de la calidad de la información inicial.

Nótese que que cuando un paciente llega al centro de atención temprana, éste ya cuenta con un cuestionario básico realizado por el médico de atención primaria en donde se recogen los motivos de su derivación al centro especializado. Posteriormente, en el centro de atención temprana, un especialista realiza una entrevista más profunda a los familiares, la cual es documentada en lenguaje natural. Posteriormente, y como resultado a esta entrevista, el paciente de clasifica dentro de uno de los siguientes procesos: físico, sensorial, cognitivo o trastornos de comportamiento.

Por tanto, los datos de los que disponemos se dividen en dos grupos:

1. Cuestionario básico con diferentes opciones, relleno por el médico de atención temprana y fácilmente manipulable por un algoritmo de aprendizaje automático.
2. Campos rellenos en lenguaje natural cumplimentados por un especialista cualificado, que entrevista a la familia del niño cuando visita el centro de atención primaria por primera vez. En estos campos, no existe consistencia en el formato, unidades, numero de registros relleno ni otros conceptos que estructuren la información. Es, por tanto, que se trata de datos que precisan de una transformación para poder usarlos en técnicas de Inteligencia Artificial.

II-C. Preparación de datos

La tercera fase está compuesta por todas las actividades para construir el dataset final partiendo de los datos en crudo. Centrándonos en el estudio actual, esta fase ha sido una de las fases más larga y tediosas debido a la naturaleza de los datos datos con los que se partían.

La preparación más compleja ha tenido que ver con el procesamiento de los campos expresados en lenguajes natural, debido a que el resto son de tipo booleano o categórico. Este procesamiento consistió en la identificación de palabras claves para detectar el sentimiento de la oración o para identificar el concepto básico expresado. Por ejemplo, atendiendo a un campo original que expresa los antecedentes familiares del paciente, buscaríamos palabras clave como: TEA (trastornos del espectro autista), (El trastorno por déficit de atención e hiperactividad), Problemas del habla, Atención, mentales, asperger o down. Para ello, descompondríamos el campo de antecedentes familiares en ocho subcampos distintos, siendo éstos los que se utilizan para el entrenamiento posterior del sistema de IA no supervisado. Este mismo proceso se sigue para el resto de campos con estas características; complicaciones en el parto, tipo de parto o complicaciones durante el parto.

Adicionalmente a lo anterior, también es necesario el preprocesamiento de algunos campos debido a la inconsistencia en las unidades empleadas. Por ejemplo, podemos encontrar campos como el del peso del recién nacido, en donde tenemos valores expresados en gramos o en kilogramos. Para su homogeneización, se automatizó el proceso de la asignación de una unidad consistente.

Igualmente, fue necesario el tratamiento de valores omisos. Para ello, se procedió a la eliminación de aquellos campos que superaran el $75\,\%$ de valores desconocidos. Para el resto, se optó por el uso de la media para los casos de variables numéricas, y la mediana para las variables categóricas.

Por último, indicar que los valores atípicos, conocidos como *outliers*, fueron descartados al considerarse que son errores a la hora de introducir los datos debido a la propia naturaleza de estos.

A modo resumen, se muestra la tabla I, en donde, además de especificar los diferentes campos que se manejan, se indican las transformaciones y descomposición que se han llevado a cabo. Con el objetivo de explicar este esquema, se detallará cada una de sus columnas atendiendo, por ejemplo, al campo de Antecedentes del paciente.

1. ***Campo de Informe Médico*** ACOIDPEAMTFAMILIARES es el identificador del campos elegido como ejemplo.
2. ***Tipo de dato***: Los posible valores serán: Categórico, Numérico, Texto libre (equivalente a lenguaje natural) o No relevante.
3. ***Relleno***: Porcentaje de valores rellenos en este campo.
4. ***Transformación***: expresa si se ha realizado alguna transformación.
5. ***Variable final***: contiene el tipo de variable final tras la transformación.
6. ***Comentario***: indica si existe algún aspecto adicional a tener en cuenta. ejemplo: No hay comentario.
7. ***S***: El valor será 1 si este campo fue escogido como variable explicativa para entrenar al modelo

Tras realizar estas tareas de preprocesado, la lista final de variables explicativas es de 40, siendo estas las empleadas en el entrenamiento del sistema de aprendizaje no supervisado del sistema.

II-D. Modelado

Durante la fase de modelado, se aplicaron diferentes algoritmos de aprendizaje así como su parametrización. Para la selección de estos algoritmos, se atendió a aquellos que han sido utilizados con éxito en aplicaciones médicas. En las siguientes subsecciones se describen éstos así como algunos de sus resultados en dicho ámbito.

II-D1. Random Forest: esta aproximación [3], denominada "Bosques Aleatorios" en español, es una combinación de árboles predictores tal que cada árbol depende de los valores de un vector aleatorio probado independientemente y con la misma distribución para cada uno de estos. Es una colección de árboles no correlacionados y de la cual promediamos los resultados. Los modelos basados en árboles de decisión son fáciles de entender ya que pueden ser representado gráficamente. Estos algoritmos han sido utilizados en campos de la medicina como por ejemplo en la predicción del riesgo de Diabetes tipo II [25].

II-D2. Regresión Lineal o ajuste lineal: El análisis de regresión lineal se utiliza para prever el valor de una variable con base en el valor de otra variable. La variable que desea prever se denomina variable dependiente. La variable que está utilizando para prever el valor de la otra variable se denomina variable independiente. Esta forma de análisis estima los coeficientes de la ecuación lineal, involucrando a una o más variables independientes que prevén mejor el valor de la variable dependiente. La regresión lineal se ajusta a una línea recta o a una superficie que minimiza las discrepancias entre los valores de salida previstos y reales. [24]. Dentro del ámbito de la medicina, podemos encontrar la aplicación de esta técnica, por ejemplo, el trabajo presentado por Anne B. Newman et al. [16] sobre la relación de los desordenes de respiración durante el sueño con los cambios de peso.

II-D3. Linear Support Vector Machine (LSVM): Son un conjunto de algoritmos de aprendizaje supervisado desarrollados por Cortes y Vapnik [4]. Estos métodos están propiamente relacionados con problemas de clasificación y regresión. Dado un conjunto de ejemplos de entrenamiento (de muestras) podemos etiquetar las clases y entrenar una SVM para construir un modelo que prediga la clase de una nueva muestra. Intuitivamente, una SVM es un modelo que representa a los puntos de muestra en el espacio, separando las clases en dos espacios lo más amplios posibles mediante un hiperplano de separación. Este hiperplano es definido como el vector entre los dos puntos, de las dos clases, más cercanos al que se

Cuadro I
DATOS EXTRAIDOS DEL SISTEMA / TRANFORMACIÓN / SELECCIÓN

Campos de Informe del médico de Atención primaria	Tipo de Dato	% Relleno	Transformación	Variable final	Comentario	S
ACO_CK_P_DERIVAR	Boolean	85		Boolean		
ACO_CK_P_REALIZAR_VI	Boolean	85		Boolean		
ACO_CK_P_SEGUIMIENTO	Boolean	85		Boolean		
ACO_CK_PAÑAL_24	Boolean	85		Boolean		1
ACO_CK_PAÑAL_NOCHE	Boolean	85		Boolean		1
ACO_CK_PAÑAL_SUCIO	Boolean	85		Boolean		1
ACO_CK_RET_MADURATIVO_LENGUA	Boolean	85		Boolean		1
ACO_CK_RETRASO_COGNITIVO	Boolean	85		Boolean		1
ACO_CK_RETRASO_MOTOR	Boolean	85		Boolean		1
ACO_CK_RETRASO_PSICO	Boolean	85		Boolean		1
ACO_CK_SEÑALES_DE_ALERTA	Boolean	85		Boolean		1
ACO_CK_TRASTORNO_COGNITIVO	Boolean	85		Boolean		1
ACO_CK_TRASTORNO_LENGUA	Boolean	85		Boolean		1
ACO_CK_TRASTORNO_COMUNICA	Boolean	85		Boolean		1
ACO_CK_TRASTORNO_MOTOR	Boolean	85		Boolean		1
ACO_CK_TRASTORNO_PSICO	Boolean	85		Boolean		1
ACO_CK_TRASTORNO_SENSORIAL	Boolean	85		Boolean		1
ACO_CMB_MEDICOS_SERVICIO	No Relevante	85			No relevante	
ACO_ID_ALERGIAS	No Relevante	0,1			Apenas se rellena	
ACO_ID_AP_ALIMENTACION	Lenguaje Natural	57	Verdaro (Come bien) Falso (Come Mal)	Boolean		1
ACO_ID_AP_COMPLICA	Texto libre	67	Respiratorias(itis)	Tres variables Boolean		1
			Convul			1
			Cardio			1
			Otra			1
ACO_ID_AP_HISTORIA_FAMI	Texto libre	8	Separación	Boolean		1
ACO_ID_AP_HOSPITALIZA	Texto libre	56	Boolean	Boolean		1
ACO_ID_AP_TIPO_LACTANCIA	Texto libre	68	Lactancia Maternal	Numerico	Hasta cuando, numerico	1
ACO_ID_CENTRO_SALUD	Categorica	85		Categorica		1
ACO_ID_DF_ACTITUD_DIAG	No relevante	1			Apenas se rellena	
ACO_ID_DF_COINCIDENC_PROB	No relevante	10			Apenas se rellena	
ACO_ID_DF_MOTIVA_COLABORAR	No relevante	1			Apenas se rellena	
ACO_ID_DF_NECES_APOYO	No relevante	0,2			Apenas se rellena	
ACO_ID_DF_REL_FAMI	Texto libre	23	Buena / Mala	Boolean		1
ACO_ID_EMBARAZO	Texto libre	76	Problemas/Sin Problemas	Boolean		1
ACO_ID_FECHA_ACOGIDA	No relevante	82			No relevante	
ACO_ID_FECHA_DERIVACION	No relevante	70			No relevante	
ACO_ID_HE_CEFALICO	Texto libre	13	Consistencia de unidades	Numerico	Apenas se rellena	
ACO_ID_HE_BIPEDESTA	Texto libre	4	Consistencia de unidades	Numerico	Apenas se rellena	
ACO_ID_HE_ESFINTERES	Texto libre	18	Consistencia de unidades	Numerico	Apenas se rellena	
ACO_ID_HE_FRASE	Texto libre	6	Consistencia de unidades	Numerico	Apenas se rellena	
ACO_ID_HE_GATEO	Texto libre	38	Consistencia de unidades	Numerico	Apenas se rellena	
ACO_ID_HE_INI_MARCHA	Texto libre	42	Consistencia de unidades	Numerico	Apenas se rellena	
ACO_ID_HE_PRI_PALABRA	Texto libre	45	Consistencia de unidades	Numerico	Apenas se rellena	
ACO_ID_HE_MARCHA	Texto libre	15	Consistencia de unidades	Numerico	Apenas se rellena	
ACO_ID_HE_SEDESTACION	Texto libre	11	Consistencia de unidades	Numerico	Apenas se rellena	
ACO_ID_MEDICO_FAMILIA	No relevante	77			No relevante	
ACO_ID_P_A_TERMINO	No relevante	14			No relevante	
ACO_ID_P_CON_ANR	No relevante	2			No relevante	
ACO_ID_P_DERIVAR_A	No relevante	0			No relevante	
ACO_ID_P_MOTIVO_ALTA	No relevante	0			No relevante	
ACO_ID_P_MULTIPLE	No relevante	0			No relevante	
ACO_ID_P_OBSERVACIONES	Dificil de extraer	0	Transformación compleja		No relevante	
ACO_ID_P_PREMATURO	Texto libre	17	Verdadero/falso	Boolean		1

llama vector soporte. Cuando las nuevas muestras se ponen en correspondencia con dicho modelo, en función de los espacios a los que pertenezcan, pueden ser clasificadas a una o la otra clase. Una buena separación entre las clases permitirá una clasificación correcta.Aplicaciones de esta aproximación podemos encontrarla, por ejemplo, en el trabajo desarrollado por Guyon et al. [8], donde realiza una selección de genes relevantes en el desarrollo del cancer de manera que puedan mejorar los diagnósticos genéticos.

II-D4. Clasificador C5: es un tipo de árbol de decisión [17] en donde dividen la muestra en función del campo que ofrece la máxima ganancia de información. Las distintas submuestras definidas por la primera división se vuelven a dividir, por lo general basándose en otro campo, y el proceso se repite hasta que resulta imposible dividir las submuestras de nuevo. Por último se vuelven a examinar las divisiones del nivel inferior, y se eliminan o podan las que no contribuyen significativamente con el valor del modelo.

Uso de este clasificador en el ámbito de la medicina podemos encontrarlo en el trabajo de [1] donde es empleado para la predicción de arteriopatía coronaria.

II-D5. CHAID: Chi-square automatic interaction detection: presentado por Kass [9], consiste en utilizar variables categóricas para ir segmentando en grupos cada vez más

pequeños nuestra población, de acuerdo a las categorías del mejor predictor. Es una técnica de árboles de decisión que usa estadísticos Chi-Cuadrado para medir la fortaleza de la asociación entre un predictor dado y el criterio que se busca maximizar, presentando además los resultados del modelo en un diagrama de árbol fácil de interpretar. Los árboles de decisión se representan bajando desde el nodo raíz hacia los nodos hoja o terminales. La raíz o nodo superior del árbol, representa la base total de datos del modelo. Esta aproximación crea un primer nivel de nodos de decisión, muestra los valores del predictor más fuerte de la variable dependiente y automáticamente determina cómo agrupar los valores de este predictor en un número manejable de categorías. Posteriormente crea otro nivel de nodos de decisión mediante el uso del predictor más fuerte entre los predictores restantes, y así continúa bajando hasta que se generen los nodos de decisión terminales del árbol.

Un ejemplo de uso de este algoritmo en medicina podemos encontrarlo en el trabajo presentado por Pavlína [10], donde propone la evaluación automática de pruebas diagnósticas.

II-D6. XGBOOST o eXtreme Gradient Boosting: esta aproximación, realizada por Chen y Guestrin [21], es parte de la familia de árboles de decisión. Es un método cuyo objetivo principal es reducir el sesgo y la varianza. Para ello inicia con la creación de árboles débiles, de manera que secuencial se vayan generando árboles nuevos (o alumno) que se centren en la debilidad (datos mal clasificados) del anterior. Después de la agregación de un alumno débil, las ponderaciones de los datos se reajustan, lo que se conoce como "reponderació". El conjunto forma un modelo sólido después de la convergencia debido a la auto-corrección después de cada nuevo alumno agregado. El algoritmo puede realizar la poda de árboles para eliminar ramas con baja probabilidad. La función de pérdida del modelo tiene un término para penalizar la complejidad del modelo con regularización para suavizar el proceso de aprendizaje (disminuir la posibilidad de sobreajuste).

Dentro del ámbito de la medicina, podemos encontrar la aplicación de esta técnica, por ejemplo, en los trabajos presentados por Liu et al. [13] o Wei y Mooney [23] para predecir la progresión del cancer de pecho o la detección de ataques epilépticos en electroencefalogramas clínicos, respectivamente,

III. Experimentación

El análisis de nuestra propuesta, que incluye el desarrollo de un MVP (Minimum Viable Product), se desarrolló en el seno del hospital San Juan de Dios (Sevilla). Concretamente en la unidad de atención temprana donde se tratan pacientes de entre 0 y 6 años con el objetivo de que desarrollen las capacidades de los pacientes para que adquieran habilidades necesarias para su integración en la sociedad. Para tal fin, se entrenaron los modelos descritos en la sección II-D atendiendo a los 420 expedientes que el hospital posee.

La bondad de estos, resumida en la tabla II, fue calculada comparando el tratamiento que propone el algoritmo de manera automática con el tratamiento que propone el experto, realizando una división de test-training de $80-20$. Nótese que el campo "Nº Vbles", indica el número de variables explicativas usadas en cada uno de los modelos. Como se explicó en el apartado II-C, el número inicial de variables elegida fueron 40. Sin embargo, para los métodos de CHAID y C5 se utilizaron 10 y 9, respectivamente, ya que las demás variables no aportaban valor al resultado.

Cuadro II
Precisión de los algoritmos de IA considerados

Modelo	**Precisión**	**Nº Vbles**
XGBoost	86.45	40
Random Forest	79.44	40
Linear Regresion	73.83	40
LSVM	68.46	40
C5	61.45	9
CHAID	68.46	10
NEURAL NETWORK	48.131	40

Atendiendo a la precisión de las diferentes aproximaciones, podemos apreciar que XGBOOST presenta el mejor comportamiento, llegando a un valor de precisión de $86{,}45$. De esta forma, podemos indicar que de los 420 expedientes analizados, se obtuvieron 370 diagnósticos correctos y 58 erróneos.

Debido a la naturaleza del estudio, donde es importante que no exista una concentración excesiva de errores en alguno de los diagnósticos, se presenta la matriz de confusión de XGBOOST en la tabla III. En esta los diferentes diagnósticos (procesos), se encuentran codificados del I al IV atendiendo a la descripción realizada en la sección II-A. Atendiendo a los valores presentes en ésta, podemos apreciar que los errores no están desbalanceados. Parte de los errores del sistema de clasificación son confusiones entre un trastorno cognitivo y un trastorno de comunicación y lenguaje, cosa que es debida a que a ciertas edades es difícil discernir entre estas dos calificaciones incluso para un médico especialista experimentado.

Cuadro III
Matriz de confusión

Real \ Predicción	I	II	III	IV
I (Cognitivo)	129	18	1	1
II (Comunicacion y Lenguaje)	8	121	2	4
III (Sensoriomotor)	10	1	73	0
IV (Socio-Comunicativo)	6	7	0	47

Por último, en la Figura1 se muestra la ganancia del sistema atendiendo a [7] y [20]. En esta se recogen dos líneas, una que indica la aleatoriedad y otra que codifica la ganancia obtenida por la técnica seleccionada. Así, podemos afirmar que nuestra propuesta cumple con los objetivos marcados, resultando de utilidad en el diagnóstico y asignación de tratamiento de los pacientes estudiados.

III-A. Depliege

Una vez entrenado el modelo y comprobada la viabilidad de este, el siguiente paso es su integración en el sistema real. El desarrollo de este paso, que se encuentra en fase

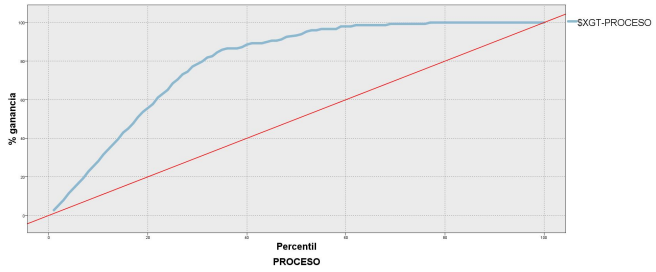


Figura 1. Ganancia del sistema.

inicial, consiste en desplegar el modelo en un sistema Cloud, de forma que sea accesible por los diferentes Centros que posee el Hospital San Juan de Dios. El consumo de este modelo vendría dado por una llamada a una API en un JSON a un micro servicio web que contuviera el valor de las cuarenta variables explicativas descritas en la sección II-C. Este servicio, respondería con la predicción de la clasificación integrada en otro JSON. Con esta arquitectura basada en un micro servicio, facilitaríamos su integración en su flujo de trabajo al reducir el proceso completo a una única llamada.

IV. Conclusiones y Trabajos Futuros

En este artículo se propone una primera aproximación del uso de técnicas de Inteligencia Artificial en el ámbito de la Atención Temprana para la ayuda a la toma de decisión de los expertos. Concretamente, se ha propuesto la aplicación de diferentes técnicas para clasificar a pacientes reales en diferentes tratamientos. El conjunto de datos así como los resultados obtenidos han sido suministrados y validados por el hospital San Juan de Dios, en donde se evidencia la utilidad de este tipo de técnicas en este ámbito.

Tras las pruebas realizadas podemos concluir que este tipo de sistema pueden ser de gran ayuda al diagnostico de aquellos ámbitos donde el diagnostico necesite de un equipo multidisciplinar o de un especialista muy experimentado. Prueba de ello, es que se pondrá en explotación el despliegue descrito en el aparatado durante el presente año.

Como línea de trabajos futuros proponemos el análisis del comportamiento en la predicción si incidiéramos en la fase de recogida de los datos. Como es habitual en el ámbito de la medicina, el conjunto de datos de partida suele contener información en lenguaje natural, lo que dificulta su procesamiento. Proponemos, con la ayuda de hospital San Juan de Dios, es la implementación de un sistema de toma de datos que, aunque permitiera un apartado en lenguaje natura donde el médico anotara apreciaciones, la idea general quedase reflejada en un solo concepto o dato numérico.

Adicionalmente, se pretende analizar la viabilidad del uso de la IA dentro del ámbito de Atención Temprana para predecir el tiempo de tratamiento de un paciente atendiendo a cuestionarios realizados en el desarrollo de este, o la identificación de trastornos cognitivos tipo TEA (Trastorno de Espectro Autista) mediante el uso de imágenes.

Referencias

[1] Ehsan Ahmadi, Gary R. Weckman, and Dale T. Masel. Decision making model to predict presence of coronary artery disease using neural network and c5.0 decision tree. *Journal of Ambient Intelligence and Humanized Computing*, 9(4):999–1011, 2018.

[2] Ana Azevedo and Manuel Filipe Santos. Kdd, semma and crisp-dm: a parallel overview. In Ajith Abraham, editor, *IADIS European Conf. Data Mining*, pages 182–185. IADIS, 2008.

[3] Leo Breiman. Random forests. *Machine Learning*, 45(2001), 2001.

[4] C. Cortes and V. Vapnik. Support vector networks. *Machine Learning*, 20:273–297, 1995.

[5] Thomas Davenport and Ravi Kalakota. The potential for artificial intelligence in healthcare. *Future healthcare journal*, 6(2):94–98, 06 2019.

[6] European Agency for Development in Special Needs Education. Atención Temprana Análisis de la situación en Europa Aspectos clave y recomendaciones. 2005.

[7] IBM Corp. Ibm spss statistics for windows.

[8] Stephen Barnhill Vladimir Vapnik Isabelle Guyon, Jason Weston. Gene selection for cancer classification using support vector machines. *Machine Learning*, 2002.

[9] G. V. Kass. An exploratory technique for investigating large quantities of categorical data. *Journal of The Royal Statistical Society Series C-applied Statistics*, 1980.

[10] Pavlína Kuráňová. Evaluation of the phadiatop test results using chaid algorithm and logistic regression. In *2015 International Conference on Information and Digital Technologies*, pages 178–182, 2015.

[11] Guang-Di Liu. A Brief Review of Artificial Intelligence Applications and Algorithms for Psychiatric Disorders. *Engineering*, 6:462 – 467, 2020.

[12] Guang-Di Liu, Yu-Chen Li, Wei Zhang, and Le Zhang. A brief review of artificial intelligence applications and algorithms for psychiatric disorders. *Engineering*, 6(4):462–467, 2020.

[13] Pei Liu, Bo Fu, Simon X. Yang, Ling Deng, Xiaorong Zhong, and Hong Zheng. Optimizing survival analysis of xgboost for ties to predict disease progression of breast cancer. *IEEE Transactions on Biomedical Engineering*, 68(1):148–160, 2021.

[14] AsunciÃMonsalve GonzÃ¡lez and Faustino NÃºÃ Batalla. La importancia del diagnÃe intervenciÃtemprana para el desarrollo de los niÃsordos: Los programas de detecciÃprecoz de la hipoacusia. *Psychosocial Intervention*, 15:7 – 28, 00 2006.

[15] Andy M.Y.Tai. Machine learning and big data: Implications for disease modeling and therapeutic discovery in psychiatry. *Artificial Intelligence In Medicine*, 9(2019), 2019.

[16] Anne B. Newman, Greg Foster, Rachel Givelber, F. Javier Nieto, Susan Redline, and Terry Young. Progression and Regression of Sleep-Disordered Breathing With Changes in Weight: The Sleep Heart Health Study. *Archives of Internal Medicine*, 165(20):2408–2413, 11 2005.

[17] J. R. Quinlan. Induction of decision trees. *Machine Learning*, 1(1):81–106, 1986.

[18] Hema Sekhar Reddy Rajula. Comparison of conventional statistical methods with machine learning in medicine: Diagnosis, drug development, and treatment. *Medicina Journal · September 2020*, 8(56), 2020.

[19] Aarón Fernández del Olmo Rocio Carrasco-Villalón, Lucía Ponce-Rodriguez. Propuesta y aplicación de una gestión por procesos para la intervención y atención temprana. *En proceso de publicación*, 2021.

[20] Jesus Salcedo and Keith McCormick. *IBM SPSS Modeler Essentials: Effective Techniques for Building Powerful Data Mining and Predictive Analytics Solutions*. Packt Publishing, 2017.

[21] Carlos Guestrin Tianqi Chen. Xgboost: A scalable tree boosting system. *Journal of The Royal Statistical Society Series C-applied Statistics*, 2016.

[22] UNESCO. Declaración de Salamanca y marco de acción sobre Necesidades Educativas Especiales. 1994.

[23] L. Wei and C. Mooney. Epileptic seizure detection in clinical eegs using an xgboost-based method. In *2020 IEEE Signal Processing in Medicine and Biology Symposium (SPMB)*, pages 1–6, 2020.

[24] Sanford Weisberg. Applied linear regression. 3.a ed. 2005.

[25] Weifeng Xu. Risk prediction of type ii diabetes based on random forest model. *2017 Third International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEICB)*, 2017.

Un sistema de apoyo al experto en actividad física basado en Metaheurísticas

Joaquín Roiz-Pagador — *Universidad Pablo de Olavide, Sevilla, España* — *Email: jroipag@alu.upo.es*
Roberto Ruiz — *Universidad Pablo de Olavide, Sevilla, España* — *Email: rruisan1@upo.es*
África Calvo-Lluch — *Universidad Pablo de Olavide, Sevilla, España* — *Email: acallu@upo.es*
Adán González-Reina — *Universidad Pablo de Olavide, Sevilla, España* — *Email: agonrei@alu.upo.es*

***Resumen*—Se presenta un sistema de apoyo, enfocado a los estilos de vida saludables, donde un sistema de información web permitirá a los profesionales en Acondicionamiento Físico, o monitores, realizar un seguimiento de la actividad física en el ámbito laboral. El sistema da soporte proponiendo una serie de ejercicios a realizar por el trabajador según el perfil patológico al que pertenezca. Estos trabajadores marcarán inicio y final de la realización de los ejercicios, además de una valoración o feedback final. Por último, se ha realizado un estudio tras una puesta en producción de 8 semanas, procesando un histórico de datos y pudiendo mejorar las recomendaciones a través del aprendizaje supervisado. En la primera parte, para ofrecer un conjunto de ejercicios a realizar, se ha escogido un algoritmo Memético compuesto por un algoritmo Genético con la Búsqueda Tabú. Posteriormente, se ha logrado llegar al 94.37 % de Accuracy a la hora de predecir series y repeticiones para un ejercicio dado.**

1. Introducción

A lo largo de los años, diferentes empresas han realizado una labor de investigación, acercándose a las diferentes estrategias existentes para minimizar las bajas laborales. Desde el punto de vista de un estilo de vida saludable, un trabajador pasa muchas horas en su puesto de trabajo, lo cual puede ocasionar ciertas lesiones que, unidas con un estilo de vida sedentario, pueden acabar en una baja laboral. Siendo éste el punto sobre el que apoyarse, se ha desarrollado un sistema que permite al especialista en ejercicio físico asignar a los trabajadores una serie de planes, acordes a su estado físico, listado de patologías y evolución a lo largo de las sesiones que se vayan teniendo. El experto en actividad física, partiendo de las patologías e historial del trabajador, podrá solicitar al sistema un conjunto de ejercicios y hacer un seguimiento para una correcta toma de decisiones.

Sistemas existentes en el mercado hacen uso de la minería de datos para extraer conocimiento y realizar toma de decisiones a largo plazo. Para casos donde los datos tienen una etiqueta definida, el aprendizaje supervisado realizará una predicción, sea clasificativa o regresiva.

En este trabajo se ha propuesto un sistema de apoyo que realice una búsqueda de ejercicios adecuados a través del algoritmo Memético, compuesto por el Genético y la Búsqueda Tabú.

Se ha realizado una prueba piloto en una empresa del sector del acero, recopilando datos de todo el proceso. Con los datos generados se ha realizado un procesamiento, generando 8 columnas y 637 filas para posteriormente aplicar un algoritmo de clasificación que seleccione las series y repeticiones adecuadas para cada ejercicio recomendado.

El siguiente documento se organiza comenzando con una breve revisión de trabajos relacionados en la sección 2, seguida de metodología, presentada en la sección 3, donde se describe el sistema y procedimientos seguidos. La sección 4 muestra las pruebas y resultados obtenidos. Por último, las conclusiones y trabajos futuros serán comentados en la sección 5.

2. Trabajos relacionados

Existen en el mercado cantidades de sistemas que ofrecen apoyo a sus usuarios. Uno de los sistemas expertos más conocidos, el sistema C Language Integrated Production System (CLIPS) [1], fue utilizado para ofrecer una solución heurística,en particular, una recomendación, adecuada a los estilos de vida saludables [2]. Dentro de aplicaciones que adaptan los estilos de vida saludables, Miveri [3] ofrece una monitorización sencilla para planificación y realización de ejercicios; otras, como la aplicación asociada a las famosas pulseras de la marca Xiaomi, MiFit [4], que monitorizan los ejercicios para sacar estadísticas, y, en algún caso, recomedaciones.

En libros, como por ejemplo en los trabajos de I. Witten [5], Jiawei Han [6] o Max Bramer [7], se tratan técnicas de procesado de datos, aprendizaje supervisado y no supervisado que han servido de ayuda para implementar el sistema propuesto.

3. Metodología

En la Figura 1 se describe conceptualmente cómo un usuario del sistema, con sus datos personales, puede ser un trabajador o un monitor. Del mismo modo, un trabajador tiene perfiles patológicos, almacenados a modo de histórico, donde el último valor añadido será el vigente. Estos perfiles

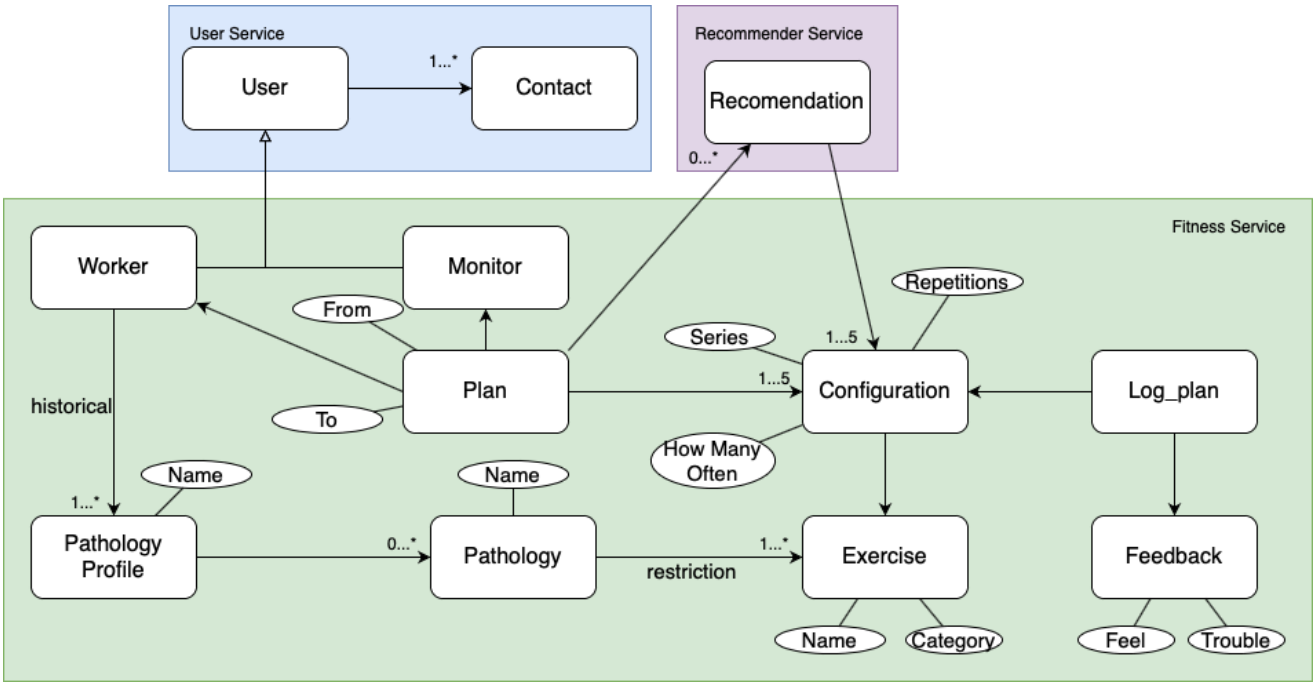


Figura 1. Diseño del modelo de datos.

patológicos están compuestos por cero (cuando no se tiene patología), una o varias patologías a la vez, que a su vez tendrán restricciones sobre ejercicios que se puedan realizar. Por otra parte, un monitor creará un plan de ejercicios que realizará un trabajador en un rango de fechas escogido, con un número de series y repeticiones para cada ejercicio en particular. Para la elaboración del plan, el monitor se podrá apoyar en el sistema solicitando una recomendación. Posteriormente se crearán una serie de registros temporales cuando estos ejercicios sean ejecutados por el empleado, a los que se asociará un feedback tras terminar el entrenamiento.

La necesidad de un sistema fácilmente escable no condiciona a delegar el despliegue de servicios en la nube. Este camino permite, a cambio de una cuantía variable, evitar un gran desembolso en la creación de servidores físicos, teniendo que ubicarlos, configurarlos y protegerlos adecuadamente. Empresas varias, como son Amazon Web Services [8], Google Cloud [9] o Azure [10], entre otros, facilitan esta labor. Sus servidores virtuales están, además, preparados para levantar orquestaciones basadas en Kubernetes [11] o Docker Swarm [12], siendo este último el utilizado en esta implementación y permitiendo un rápido y controlado despliegue, replicado y monitorización con sofwares como Swarmpit [13].

En la Figura 2 se describe lo anteriormente comentado. Este sistema basado en microservicios ofrece un servicio de registro y login (Auth Service), validación de permisos (Authorization Service), gestión de usuarios (User Service), gestión fitness (Fitness Service), registro de logs (Record Service) y sistema de recomendación (Recommender Service). Cada uno de estos conecta con una o varias bases de datos adecuadas al modelo con el que trabaja. También son mostrados el servicio de balanceo de carga (Discovery Server) y la el servicio o API Gateway (Zuul Service), encargado de filtrar las peticiones entrantes.

3.1. Sistema de Apoyo al experto

El principal motivo de este sistema de apoyo radica en el punto de vista del experto en la actividad física y del deporte, donde esta persona, mediante un sistema web, asigna una batería de ejercicios dispuesta de forma ordenada, con series y repeticiones que realizar un número de veces durante un período de tiempo. El proceso se dividirá en dos partes, una primera donde el sistema realizará una búsqueda de aquellos ejercicios que puedan ser realizados, aplicando un filtro de restricciones patológicas y ordenando según el feedback obtenido de los ejercicios efectuados con anterioridad. Posteriormente, en la segunda parte, se aplica un algoritmo de búsqueda que sea capaz de encontrar, en un tiempo razonable, un subconjunto de ejercicios adaptado al trabajador, teniendo también en cuenta el número de veces que el usuario ha realizado esos ejercicios anteriormente y el feedback derivado.

Como primer paso para realizar una propuesta, el microservicio recomendador (recommender-service) recibe una petición a través de una interfaz web (API REST), la cual, gracias al identificador del trabajador al que se le quiere adaptar el plan, solicita los ejercicios al microservicio correspondiente. En la tabla 1 puede observarse la estructura de los datos que este microservicio extrae para un trabajador

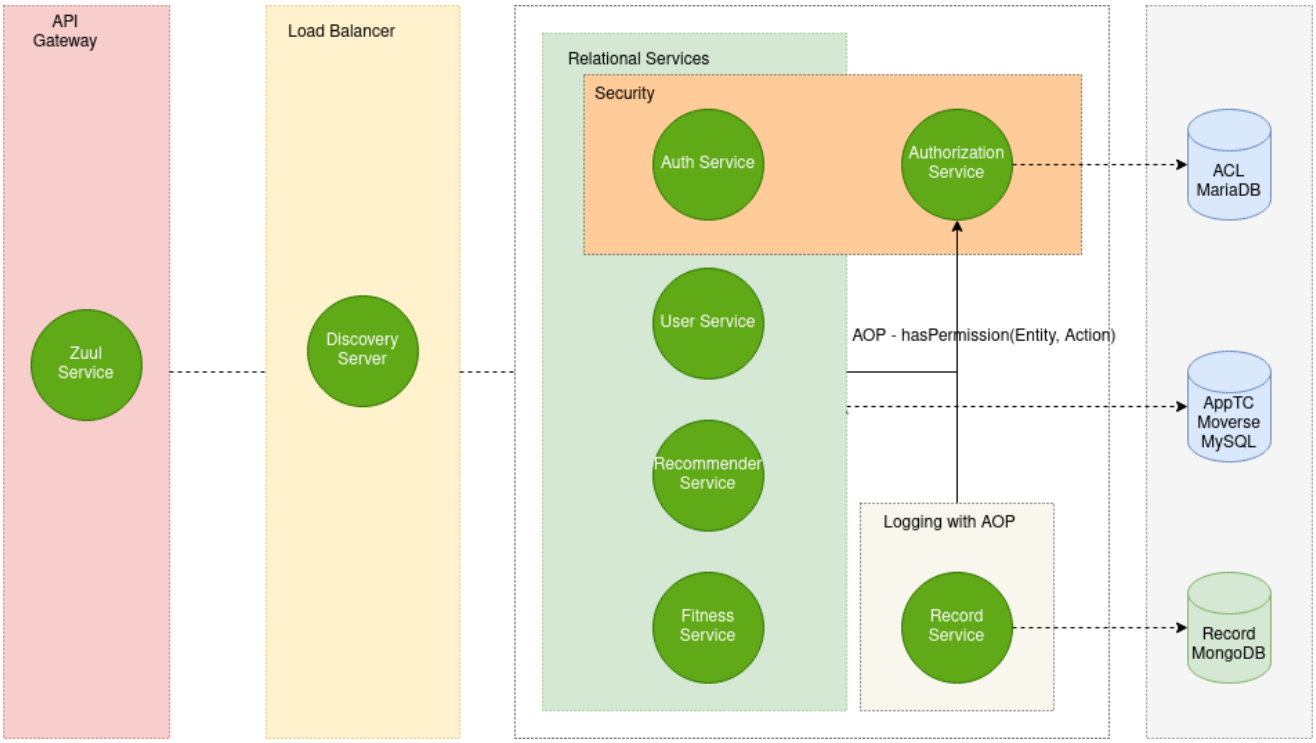


Figura 2. Diseño de la arquitectura de microservicios.

según sus patologías vigentes. Tras realizar este filtrado, surgen varias posibilidades:

1. Que no existan datos históricos sobre la realización de ejercicios en el sistema. Se devuelve un subconjunto de entre 50 y 80 ejercicios estratificados según la categoría a la que pertenece cada ejercicio repartidos equitativamente. Se ha decidido el tamaño 50-80 para optimizar los recursos en memoria a la hora de reducir el tiempo y la carga computacional.
2. Que el trabajador no tenga historial previo, pero sí existan históricos sobre la realización de ejercicios de otros trabajadores en el sistema. Se realiza una búsqueda de los ejercicios realizados por usuarios similares según su perfil patológico, se puntúan y ordenan según el feedback recibido mediante la Ecuación 1, donde f (feel) hace referencia a cómo se encuentra físicamente y t (trouble) a si el trabajador ha tenido algún problema a la hora de realizar el ejercicio, tomando valor 1 en el mejor de los casos y 3 en el peor. Además, se obtiene la cantidad de veces que hayan realizado los ejercicios los trabajadores. Posteriormente, se devuelve este ranking estratificado según la categoría a la que pertenece y filtrando de 50 a 80 ejercicios. En caso de no haber suficientes ejercicios tras filtrar este histórico (50 al menos), entraríamos en la situación del dato anterior, en la cual no habria histórico suficiente en el sistema.
3. Que el trabajador haya realizado anteriormente planes completos de ejercicios y exista un historial propio suficientemente grande. Se puntúan y ordenan según el feedback recibido, se obtiene la cantidad de veces que ha realizado anteriormente ese ejercicio el trabajador y se devuelve un subconjunto de entre 50 y 80 ejercicios estratificados según su categoría. En caso de no llegar a los 50, se vuelve a la primera casuística.

$$s = \frac{MAX(f) + MAX(t) - (f + t)}{MAX(f) + MAX(t) - (MIN(f) + MIN(t))} \quad (1)$$

El segundo y último paso del proceso de recomendación consiste en la selección de un subconjunto de ejercicios de la lista generada en la etapa anterior mediante un algoritmo metaheurístico. El algoritmo escogido para esta propuesta es un Memético, compuesto por un Genético y una Búsqueda Tabú, que permita explorar de forma equilibrada el espacio de soluciones existente. En esta fase se distinguen tres elementos: restricciones a la hora de crear un individuo, fitness que valore la calidad, y mutación o generación de solución vecina.

Según las restricciones obtenidas a través del experto en la actividad física y del deporte, una planificación deberá tener entre 3 y 5 ejercicios. Estos ejercicios deben ir acompañados de estiramientos de su mismo tipo, salvo caso en el que no existiera (como es el caso de los ejercicios abdominales, o core). Del mismo modo, un estiramiento

Tabla 1. DESCRIPCIÓN DEL CONJUNTO DE DATOS UTILIZADO PARA LAS RECOMENDACIONES.

Columna	Tipo	Descripción
Ejercicio	String	Nombre del ejercicio realizado.
Categoria	String	Categoría del ejercicio realizado.
Cantidad	Number	Cantidad de veces que se realizó el ejercicio anteriormente.
Score	Number	Puntuación obtenida según el feedback, en caso de haber tomado los ejercicios de un histórico existente.

debe ir acompañado de un ejercicio de la familia equivalente. Tampoco pueden existir ejercicios repetidos en las recomendaciones. Los estiramientos deberán realizarse en último lugar.

Para valorar la bondad de un individuo o solución candidata, al igual que con las restricciones, se han tenido en cuenta las indicaciones del experto en la actividad física y del deporte, dando lugar a las siguientes consideraciones:

- Un conjunto de ejercicios deberá ser lo más diverso posible, entendiendo la diversidad como la cantidad de categorías diferentes existentes en el invididuo a evaluarse.
- Un individuo cuyos ejercicios contengan una mayor puntuación (Score en la Tabla 1) tendrá mayor posibilidades de ser escogido.
- Aquellos ejercicios que se hayan realizado anteriormente tendrán menos probabilidad de ser seleccionados, permitiendo una mayor diversificación entre ejercicios.
- La calidad de una recomendación con 3 ejercicios deberá ser equivalente a una recomendación con 5 ejercicios.

La ecuación 2 recoge lo anterior, buscando maximizar F.

$$F = \frac{d}{t+c} * (1+s) \tag{2}$$

Donde,

- d es la diversidad de categorías escogidas.
- c es el sumatorio de veces que han realizado los ejercicios escogidos
- t es el número de ejercicios escogidos.
- s es el score, o puntuación obtenida por el ejercicio en historiales previos (0, en caso de no existir este valor o ser una valoración negativa).

Tanto en la generación de soluciones vecinas en búsquedas locales como en las mutaciones del genético se hará uso del mismo procedimiento:

- 50 % de probabilidades de sustituir un ejercicio por otro que no exista en el individuo.
- En caso de no realizar esta sustitución, existen dos posibilidades:
  - Si se puede, añadir un ejercicio más al listado.
  - Si se puede, eliminar un ejercicio del listado aleatoriamente.

Tabla 2. DESCRIPCIÓN DEL CONJUNTO DE DATOS PARA SU POSTERIOR USO EN LA EXPERIMENTACIÓN.

Columna	Type	Descripción
Edad	Number	Edad del trabajador.
Código Postal	String	Código postal de la zona donde reside el trabajador.
Repeticiones	Number	Repeticiones del ejercicio realizado.
Series	Number	Series del ejercicio realizado.
Perfil Patológico	String	Perfil patológico asociado al trabajador en el momento de crear el plan.
Categoria	String	Categoría del ejercicio realizado.
Ejercicio	String	Nombre del ejercicio realizado.
Registro ejecución	DateTime	Fecha en la que se registró la realización del ejercicio.

3.2. Mejora del Apoyo, extracción y procesado de datos

Tras un tiempo de ejecución, la aplicación ha conseguido almacenar una serie de datos históricos que pueden ser interesantes a la hora de realizar toma de decisiones, así como una mejoría del sistema de apoyo existente. Uno de los caminos posibles a aplicar es el aprendizaje supervisado, donde algoritmos de clasificación puedan elegir un conjunto de ejercicios, series y repeticiones para el trabajador. Este último proceso será descrito en la Sección 4.

Para realizar esta labor de apoyo, se ha elaborado una primera aproximación extrayendo los datos después de una prueba de 8 semanas en producción.

Para el tratamiento de datos antes de la experimentación, se ha hecho uso de la herramienta Knime [14] de análisis y procesamiento de datos. En ella, y como paso necesario para obtener un conjunto de datos de calidad para su posterior análisis, se ha hecho uso de un preprocesado. En primer lugar, a la hora de extraer los datos en un CSV se han anonimizado los datos personales.

Los datos extraídos contienen valores missing en algunos casos, como aquellos trabajadores que comenzaron un ejercicio y no lo terminaron o no respondieron al feedback tras completar su entrenamiento. Debido a la cantidad de registros existentes para estos casos, se ha tomado la decisión de eliminarlos.

Posteriormente, se eliminaron registros con ejercicios cuyo tiempo de duración superara los 40 minutos. Finalmente, los 7 atributos seleccionados manualmente fueron el código postal, el perfil patológico, la edad, la categoría, el ejercicio, las series, repeticiones y registro (en formato fecha), descritos en la Tabla 2.

4. Experimentación

A continuación se mostrarán dos análisis diferentes, describiendo en primer lugar la elección de la metaheurística utilizada para la selección de ejercicios del sistema en producción. Posteriormente, se comentarán las predicciones realizadas con los datos extraídos del sistema tras un tiempo de ejecución.

4.1. Selección de Metaheuristica

Con el fin de realizar una selección correcta del algoritmo de búsqueda que realice la recomendación, se han realizado varias ejecuciones de diferentes algoritmos [15], haciendo uso de un subconjunto de ejercicios de tamaño 80 (como fue mencionado en la sección anterior), escogidos de forma aleatoria y estratificada. El algoritmo Memético supera con creces en el fitness obtenido. Como puede observarse en la Tabla 3, la combinación de genético y tabú ha obtenido una mejora considerable de estos dos por separado. Ante la evidente mejor solución alcanzada por el algoritmo Memético, se ha utilizado en producción.

Tabla 3. COMPARATIVA ENTRE LAS DIFERENTES METAHEURÍSTICAS PROBADAS.

Algoritmo\ Prueba	Genético	Memético	Búsqueda Tabú	Enfriamiento Simulado	Hill Climbing
1	0.60	0.75	0.57	0.30	0.50
2	0.60	0.75	0.60	0.42	0.42
3	0.50	0.60	0.42	0.57	0.42
4	0.42	0.75	0.60	0.30	0.66
5	0.60	0.75	0.60	0.33	0.16
MEDIA	0.54	0.72	0.56	0.38	0.43

4.2. Clasificación y validación

Tras la limpieza comentada en la Sección 3.2, 637 filas dan lugar a una posibilidad de aprendizaje interesante. Se ha hecho uso del clasificador Random Forest [16] como punto de partida y validación Hold Out estratificada según la clase, repartiendo en 70 % para entrenamiento y 30 % para test. A su vez, se ha dividido el conjunto de entrenamiento en entrenamiento y validación, particionando en 70 % y 30 % nuevamente para elegir el número de árboles escogido, fijado finalmente a 100. Los diferentes puntos de vista, o clases seleccionadas para predecir, serán el ejercicio (en este caso, se eliminaría la categoría y las series y repeticiones del conjunto de datos dado), el conjunto <serie, repetición>, la serie y la repetición.

Después de la ejecución, se ha obtenido 84.37 % de Accuracy para la clase <serie, repetición>, 71.88 % de Accuracy en el caso de predicción de ejercicio.

Por otro lado, la clasificación por separado de las series y repeticiones, creando un modelo para cada una de estas categorías anteriores, sería capaz de encontrar el 90.63 % y 60 % de Accuracy en series y repeticiones, respectivamente. Esto plantea la necesidad de trabajar con más datos para un aprendizaje más adecuado en el caso de las repeticiones, como puede observarse.

5. Conclusiones y trabajos futuros

Con este trabajo planteado, el primer paso dado nos acerca a una realidad bastante factible, donde expertos en la actividad física y del deporte podrán agilizar su trabajo gracias a un sistema de apoyo. La aplicación ha sido probada bajo las exigencias que conlleva una prueba real de ocho semanas de duración, en las que se han realizado propuestas de ejercicios y recopilado datos de interés. El sistema de apoyo descrito será capaz de adaptarse rápidamente a nuevos contextos, como la nutrición. Del mismo modo, será capaz de modificarse de manera sencilla para una mejor valoración o fitness, una vez tenidos en cuenta nuevos datos que generen los trabajadores. Por otra parte, la propuesta realizada para la mejora del modelo a través del aprendizaje supervisado permite la adaptación de forma genérica a los datos existentes, pudiendo realizarse una automatización en un futuro para la limpieza y procesado de los datos, así como una comparativa entre los diferentes y principales algoritmos de aprendizaje supervisado existentes. La puesta en producción de esta última mejora clasificativa quedará pendiente para los siguientes meses.

Se propone una futura línea que permita la adaptación de estos modelos al aprendizaje continuo que facilite la correcta adecuación a cambios.

Algoritmos basados en diferentes medidas de correlación o unión entre los diferentes atributos podrán dar un gran nivel de conocimiento de las relaciones existentes. Esto, además de permitir una correcta selección de atributos que agilicen el procesado en un futuro donde exista una mayor cantidad de datos, podrá dar lugar a una extracción de conocimiento y posible mejora de los algoritmos que se usen, los cuales también serán motivo de estudio para esta línea.

Agradecimientos

Trabajo parcialmente financiado por el Ministerio de Economía y Competitividad, proyecto TIN2014-55894-C2, y por la Junta de Andalucía, Exp.1155595.

Referencias

[1] "C language inte-grated production system (clips)," http://www.clipsrules.net.

[2] T. Lorincz, B. Szakonyi, A. Lipovits, and I. Vassányi, "Development of an expert system framework for lifestyle improvement," 11 2017, pp. 93–98.

[3] L. Moreno, J. Angulo, R. Rueda, Álvaro Rodríguez, D. Sojo, L. Romero, V. Escribano, and I. Díaz, Miveri Health Solution, 2016, https://andalucia.openfuture.org/startups/miveri-health-solutions/.

[4] Anhui Huami Information Technology, LTD, "Mi Fit," https://play.google.com/store/apps/details?id=com.xiaomi.hm.healthhl=es, 2020, online; accedido el 13 de Mayo de 2020.

[5] I. Witten, E. Frank, M. Hall, and C. Pal, Data Mining: Practical Machine Learning Tools and Techniques, ser. The Morgan Kaufmann Series in Data Management Systems. Elsevier Science, 2016. [Online]. Available: https://books.google.es/books?id=1SylCgAAQBAJ

[6] J. Han, M. Kamber, and J. Pei. (2012) Data mining concepts and techniques, third edition. Waltham, Mass.

[7] M. Bramer, Principles of Data Mining, 3rd ed., ser. Undergraduate Topics in Computer Science. London: Springer, 2013.

[8] "Amazon web services," https://aws.amazon.com.

[9] "Google cloud platform," https://cloud.google.com.

[10] "Microsoft azure," https://azure.microsoft.com.

[11] “Kubernetes,” https://kubernetes.io.

[12] “Docker swarm,” https://docs.docker.com/engine/swarm.

[13] “Swarmpit),” https://swarmpit.io/.

[14] M. R. Berthold, N. Cebron, F. Dill, T. R. Gabriel, T. Kötter, T. Meinl, P. Ohl, C. Sieb, K. Thiel, and B. Wiswedel, “KNIME: The Konstanz Information Miner,” in Studies in Classification, Data Analysis, and Knowledge Organization (GfKL 2007). Springer, 2007.

[15] J. Roiz-Pagador, MetaheuristicsFinalWork, 2018, https://github.com/QuiniRoizPagador/MetaheuristicsFinalWork.

[16] L. Breiman, “Random forests,” Mach. Learn., vol. 45, no. 1, pp. 5–32, 2001.

StreetQR: Informative assistance device for street name plates and places of interest

Jesús Benito-Picazo, Jorge García-González, Gonzalo Ramos-Jiménez, Ezequiel López-Rubio
Department of Computer Languages and Computer Science
University of Málaga
Málaga, Spain
{jpicazo, jorgegarcia, ramos, ezeqlr}@lcc.uma.es

***Abstract*—Because of the innumerable ways it affects human beings, information is the most valuable good in nowadays' society. Consequently, we are surrounded by hundreds of devices that constantly gather and supply information about the places where people move around. One of the environments which produces a higher amount of information are the streets of our cities. Thus, our urban environments are packed with many devices intended to extract information and help pedestrians passing by. Thus, in this paper we present *StreetQR*: A new approach to urban mobility consisting of a device intended to help pedestrians by offering useful information about the spot of the city they are in, by analyzing through deep-learning technologies every object present in the environment.**

***Index Terms*—Data mining, video surveillance, deep learning**

I. INTRODUCTION

From the most rudimentary traffic lights to the most advanced Artificial Intelligence-based automatic video surveillance systems, over time we can observe an increment of electronic computerised devices populating modern urban environments. As the technology improves, these devices get more complex and sophisticated, being designed to offer an increasing amount of services to the people populating the cited environments, transforming them in the so-called *Smart Cities*. A good definition of this concept and its implications appears in [1], where the authors provide a focused and operational definition of this construct and a deep analysis of the factors that determine Smart Cities performance in Europe. Smart cities are possible thanks to some emerging new technologies such as the Internet of the Things (IoT), as it is described in [2]. In this paper, the authors present a Cloud centric vision for worldwide implementation of Internet of Things, discussing the technologies and applications of the IoT research in the near future. One of the most important approaches to the Smart Cities paradigm consists of the inter-vehicle communication. In this topic, we can find several works such as the one presented in [3], where we can find a new IEEE 802.11 based multi-hop broadcast protocol (UMB) designed to address reliability problems of multi-hop broadcast in urban areas. The work presented in [4] also approaches the problem of urban multi-vehicle communication with the presentation of the vehicular fog computing (VFC) which is an architecture consisting on a collaborative multitude of near-user edge devices to carry out communication and computation, based on the better utilization of individual communication and computational resources of vehicles. Smart Cities infrastructure does not only relies on vehicle sensors, in fact, there is a class of devices that represents a permanent and inexhaustible source of information for urban environments: smartphones. Sensor enabled mobile phones or smart-phones are the center of a next revolution in social networks, green applications, global environmental monitoring, personal and community healthcare, sensor augmented gaming, virtual reality and smart transportation systems, as it is described in [5]. Along the same lines, the work presented in [6] describes a novel framework for an Energy Efficient mobile sensing system that uses smartphone sensors to recognize user states as well as to detect state transitions. Smart Cities also get a lot of benefit from automatic video surveillance systems as they help to monitor the activities taking place in different urban environments. Therefore, systems such as the one described in [7] are very helpful insofar as it comprehends a real-time video surveillance system capable of performing the detection and tracking of people from video streams supplied by outdoor video cameras, in order to identify the activities these people are carrying out. In the same research field we can find the work described in [8], where a face identification system is presented with the capability of identifying the people accessing certain facilities by analyzing video streams installed in these facilities' entry points. Deep learning-based object detectors and identifiers are a powerful technology when it comes to video surveillance systems' construction insofar as they are capable of identifying both objects and actions performed by these objects [9]. In this field, several works have been developed. For example, in [10] the authors present a novel 3D CNN model for action recognition capable of capturing the motion information encoded in multiple adjacent frames supplied by video cameras placed in airport environments. Deep-learning based systems are also useful to perform critical tasks in crowd monitoring in urban environments as it is described in [11], where the authors provide a survey of recent Convolutional Neural Network (CNN) based approaches that have demonstrated significant improvements over earlier methods, in image crowd counting and density estimation. Traffic monitoring is also benefited from Deep Learning, as we can read in [12] and [13].

Finally, every sensor placed in any urban monitoring system



Fig. 1. Prototype overview.

represents a very important information source to be taken into account for data mining tasks that often will be used to improve information systems having a positive impact in the peoples' lives. For example, in [14] the authors present a distributed infrastructure based on wireless sensors for air pollution monitoring and mining, in urban environments. In the same field, [15] presents a method to infer air quality of any arbitrary location from environmental data and historical air quality data from sparse monitoring locations in real time. Data mining can be used to monitoring human activities for healthcare applications, as it can be found in and [16] and [17].

In this paper we present StreetQR: a new deep learning-based device designed for improving urban mobility for pedestrians by offering useful information about the place the device is placed in, while it collects data about pedestrians and automobile transit in a certain spot of the city.

The rest of the paper is organized as follows: The second section will illustrate the construction and most relevant features of StreetQR. The third section presents the deep learning-based object detection model to be used for pedestrian and vehicle classifying and counting. The last section presents the conclusions and future research lines motivated by this device's development.

II. DEVICE CONSTRUCTION

A. Overview of StreetQR

As it has been stated in Section I, in this work we present a new concept of smart device intended to serve as the base for street name plates. It also incorporates a QR code which can be scanned using a smartphone camera so that any pedestrian can obtain updated information in its smartphone about the specific spot the device is placed in. In Figure 1 we can observe StreetQR once it is installed in a street name plate pole.

The information to be supplied by StreetQR would be decided by the municipality authorities and it could consist of information about the street, such as its name and history, or it could be supplying any other information that authorities would consider important (museums, places of interest, official buildings, etc.). It also may be used as an information spot for real-time information feed about the city conditions in that street, such as state of the traffic, pollution values and any emergency information the local authority has considered to include. This information can be accessed by just scanning the QR code present in the device with a smartphone. Once this QR code is scanned, the phone's web browser redirects the user to a website specially developed to contain that information.

StreetQR also incorporates a camera in charge of capturing a video feed from the environment the device is placed in. Such video feed will be analyzed in real time using a deep learning-based detection algorithm that will be deployed in a local power efficient hardware device. The cited analysis will yield information about the pedestrian and traffic flow of a certain street which will be sent to a server placed in a city hall-dependent facility in order to be processed and analyzed by qualified data mining experts. StreetQR has one LED light that flashes at different frequencies and a speaker to warn pedestrians about any abnormal situation taking place in a certain street.

B. Physical device

In Figure 2 we can observe the main parts conforming the device from both front and back angles. These parts are briefly explained below:

- *Street name plate*. This is the place where the name of the street will be placed. It will have the format required by the municipality authority.
- *Camera*. Situated in the front side of the device, this camera will be in charge of obtaining a video stream of the street which will be fed to the detection model.
- *QR Code*. This is the QR code that, once scanned using a smartphone, will take the pedestrian to a URL where the local administration will have placed the information it considers convenient.
- *Antenna*. StreetQR has a communication system that sends anonymous information to the local administration about the pedestrian and vehicle flow in that specific spot. The external antenna improves the signal quality of the 4G communication system.
- *Led bulb*. As mentioned before, StreetQR is capable of warning the pedestrians and vehicles by using light signals. These signals are produced by this LED-diode bulb.
- *Speaker*. Just as the LED light, this device is intended to pass on any warning to pedestrians.
- *Enclosure*. A steel structure that protects the internal hardware of the device from weather and vandalism as well as supplying a solid substrate for fixing this hardware.
- *Power source*. The power source of the device, which should be connected to the general power network of the municipality.
- *Processing Hardware*. A deep learning-based system of this characteristics needs an efficient yet powerful pro-

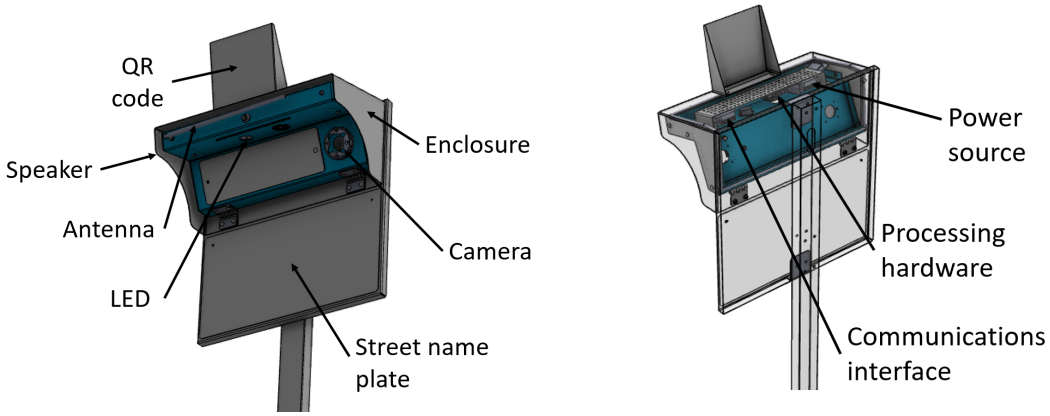


Fig. 2. Schematic of the StreetQR prototype.



Fig. 3. Information spot operation mode.

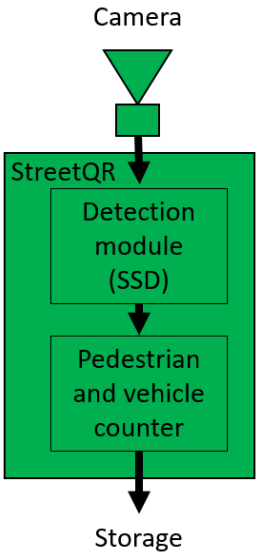


Fig. 4. Pedestrian and vehicle information gathering.

cessing device. In this case, StreetQR will incorporate a system on chip processing board.

- *Communications Interface*. 4G modem in which the bi-directional wireless communications of the device relies. The wide signal range reached by the 4G technology allows the StreetQR installation in almost any environment.

C. Device operation

Apart from being a mount for a street name plate, StreetQR has three main operation modes, according to the different uses mentioned in section II-A. As it appears in Figure 3, the first operation mode relies in its utility as an information spot. This operation mode is passive as it works just by scanning the included QR code with a smartphone's camera .

The second operation mode is the most complex from a computational point of view as it utilizes deep learning technologies to scan in real-time the video stream coming from the built-in camera in order to track and count the number of people and vehicles transiting through the street where the device is placed. Once a day, the system transmits the number of pedestrians and vehicles detected during the day to a server designated by the municipality authorities, where they can be analyzed by qualified personnel (Figure 4).

The third operation mode of StreetQR leverages its capabilities to be remotely accessed through the internet by an authorized agent who will be able to transmit in real-time any kind of warning to pedestrians by using the LED bulb and the speaker mounted on the device (Figure 5).

D. Implementation details

Software was implemented using Python[1] [18] as programming language and Tensorflow[2] [19] as deep learning framework. Libraries as CUDA[3] [20], Numpy[4] [21] and OpenCV[5]

[1]https://www.python.org/
[2]https://www.tensorflow.org/
[3]https://developer.nvidia.com/cuda-zone
[4]https://numpy.org/
[5]https://opencv.org/

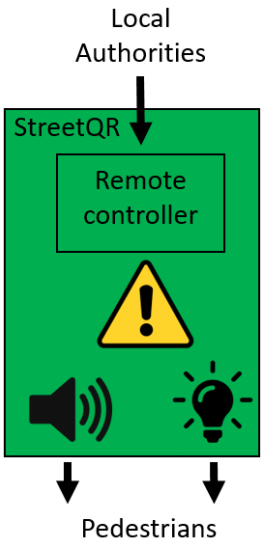


Fig. 5. Communication with pedestrians.

[22] have also been fundamental parts of development.

III. DETECTION MODEL

The proposed device key software element is a deep learning-based object detection method applied to detect persons and vehicles. StreetQR uses a Single Shot Multibox Detector (commonly known as *SSD*) [23] as object detector.

SSD is a deep neural network method designed to work with real-time processing speed. Its main difference from other slower deep convolutional neural network-based object detection models such as Faster-RCNN [24] is to dispense with the region proposal neural network. Instead, *SSD* input has a fixed image size (typically 300×300) and a multi-scale feature map strategy is applied to detect objects independently based on manually pre-selected boundary boxes shapes for each location. These boundary boxes are defined in order to match real-life objects.

VGG16 [25] is a deep convolutional neural network model to classify images in order to decide to which class belong the objects appearing in them (*Image Classification*). *SSD* applies *VGG* to extract feature maps, then 3×3 convolutional filters are applied for each location to obtain scores for each class and boundary box. Following *VGG16*, *SSD* includes 6 more convolutional layers. Due to spatial dimension reduction provoked by convolutional layers, the feature maps resolution also decreases and *SSD* uses the lower resolution feature maps in order to detect larger scale objects using the same boundary box shapes.

Finally, a total of 8732 detections per class is obtained and Non-Maximum Suppression algorithm is applied in order to delete multiple detection of the same object. Figure 6 on page 5 shows *SSD* method structure.

The *SSD* applied is a pretrained one. It was trained on COCO dataset [26], which is a public[6] dataset published by Microsoft widely used to train object detection methods due to its very useful classes and common context images.

IV. CONCLUSIONS

This paper presents a patented device designed to improve pedestrians urban mobility by offering information while collecting data about pedestrians and vehicles using deep neural networks. The device offers three operation modes as described in section II-C on page 3: a QR code allows to obtain information about the place, a camera and processing hardware are included to detect pedestrians and vehicles in order to count them and communication hardware allows the device to be accessed by an authorized agent to transmit warnings by using the built-in speaker and LED bulb.

The device is designed to be integrated to the street sign so that it is not a hindrance in the street. The device hardware cost has been carefully optimized in order to adjust it to a limited budget so many can be installed in a city.

As future work, a better device-sign integration should be achieved in order to be as discreet as possible so the city aesthetics is not affected by its installation. Fine-tuning the deep neural network used for detection and classification method would be also a way to improve the device's performance.

ACKNOWLEDGMENT

This work is partially supported by the Ministry of Economy and Competitiveness of Spain under grants TIN2016-75097-P and PPIT.UMA.B1.2017. It is also partially supported by the Ministry of Science, Innovation and Universities of Spain under grant RTI2018-094645-B-I00, project name Automated detection with low-cost hardware of unusual activities in video sequences. It is also partially supported by the Autonomous Government of Andalusia (Spain) under project UMA18-FEDERJA-084, project name Detection of anomalous behavior agents by deep learning in low-cost video surveillance intelligent systems. All of them include funds from the European Regional Development Fund (ERDF). It is also partially supported by the Autonomous Government of Andalusia (Spain) under division UMA-CEIATECH-22 from the FEDER 2014-2020, project name DIAS2P Smart devices for improving crosswalk safety. All of them include funds from the European Regional Development Fund (ERDF). It is also supported by the Smart-Campus plan of the University of Málaga. The authors thankfully acknowledge the computer resources, technical expertise and assistance provided by the SCBI (Supercomputing and Bioinformatics) center of the University of Málaga. They also gratefully acknowledge the support of NVIDIA Corporation with the donation of two Titan X GPUs used for this research. The authors acknowledge the funding from the Universidad de Málaga. *StreetQR* is protected under the Spanish office of patents. Patent Number: ES2647968B1.

[6]https://cocodataset.org/

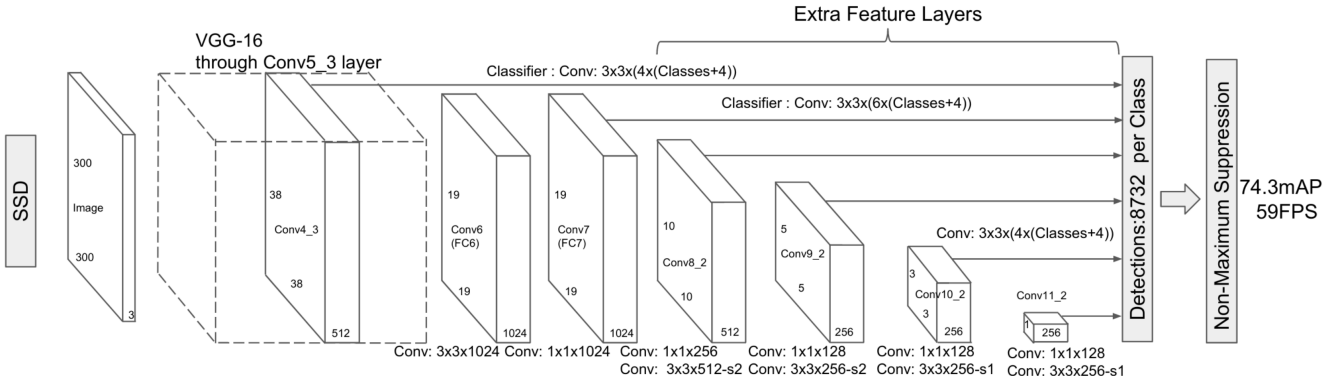


Fig. 6. *SSD* neural network structure obtained from the original paper [23].

REFERENCES

[1] A. Caragliu, C. del Bo, and P. Nijkamp, "Smart cities in europe," *Journal of Urban Technology*, vol. 18, no. 2, pp. 65–82, 2011, cited By :1341. [Online]. Available: www.scopus.com

[2] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of things (iot): A vision, architectural elements, and future directions," *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1645–1660, 2013, cited By :6194. [Online]. Available: www.scopus.com

[3] G. Korkmaz, F. Özgüner, E. Ekici, and Özgüner, "Urban multi-hop broadcast protocol for inter-vehicle communication systems," in *VANET - Proceedings of the First ACM International Workshop on Vehicular Ad Hoc Networks*, 2004, pp. 76–85, cited By :527. [Online]. Available: www.scopus.com

[4] X. Hou, Y. Li, M. Chen, D. Wu, D. Jin, and S. Chen, "Vehicular fog computing: A viewpoint of vehicles as the infrastructures," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 6, pp. 3860–3873, 2016, cited By :473. [Online]. Available: www.scopus.com

[5] W. Z. Khan, Y. Xiang, M. Y. Aalsalem, and Q. Arshad, "Mobile phone sensing systems: A survey," *IEEE Communications Surveys and Tutorials*, vol. 15, no. 1, pp. 402–427, 2013, cited By :389. [Online]. Available: www.scopus.com

[6] Y. Wang, J. Lin, M. Annavaram, Q. A. Jacobson, J. Hong, B. Krishnamachari, and N. Sadeh, "A framework of energy efficient mobile sensing for automatic user state recognition," in *MobiSys'09 - Proceedings of the 7th ACM International Conference on Mobile Systems, Applications, and Services*, 2009, pp. 179–192, cited By :344. [Online]. Available: www.scopus.com

[7] I. Haritaoglu, D. Harwood, and L. S. Davis, "W4: Real-time surveillance of people and their activities," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 809–830, 2000, cited By :2192. [Online]. Available: www.scopus.com

[8] D. Dalwadi, Y. Mehta, and N. Macwan, *Face recognition-based attendance system using real-time computer vision algorithms*, ser. Advances in Intelligent Systems and Computing, 2021, vol. 1141. [Online]. Available: www.scopus.com

[9] S. Herath, M. Harandi, and F. Porikli, "Going deeper into action recognition: A survey," *Image and Vision Computing*, vol. 60, pp. 4–21, 2017, cited By :292. [Online]. Available: www.scopus.com

[10] S. Ji, W. Xu, M. Yang, and K. Yu, "3d convolutional neural networks for human action recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 221–231, 2013, cited By :2884. [Online]. Available: www.scopus.com

[11] V. A. Sindagi and V. M. Patel, "A survey of recent advances in cnn-based single image crowd counting and density estimation," *Pattern Recognition Letters*, vol. 107, pp. 3–16, 2018, cited By :200. [Online]. Available: www.scopus.com

[12] X. Liu, W. Liu, T. Mei, and H. Ma, *A deep learning-based approach to progressive vehicle re-identification for urban surveillance*, ser. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2016, vol. 9906 LNCS, cited By :143. [Online]. Available: www.scopus.com

[13] H. Liu, Y. Tian, Y. Wang, L. Pang, and T. Huang, "Deep relative distance learning: Tell the difference between similar vehicles," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2016-December, 2016, pp. 2167–2175, cited By :274. [Online]. Available: www.scopus.com

[14] Y. Ma, M. Richards, M. Ghanem, Y. Guo, and J. Hassard, "Air pollution monitoring and mining based on sensor grid in london," *Sensors*, vol. 8, no. 6, pp. 3601–3623, 2008, cited By :123. [Online]. Available: www.scopus.com

[15] H. . Hsieh, S. . Lin, and Y. Zheng, "Inferring air quality for station location recommendation based on urban big data," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, vol. 2015-August, 2015, pp. 437–446, cited By :96. [Online]. Available: www.scopus.com

[16] A. Yassine, S. Singh, and A. Alamri, "Mining human activity patterns from smart home big data for health care applications," *IEEE Access*, vol. 5, pp. 13 131–13 141, 2017, cited By :107. [Online]. Available: www.scopus.com

[17] R. Kavitha, S. Sangeetha, and A. G. Varghese, "Human activity patterns in big data for healthcare applications," *International Journal of Innovative Technology and Exploring Engineering*, vol. 8, no. 9 Special Issue 3, pp. 1101–1103, 2019, cited By :87. [Online]. Available: www.scopus.com

[18] G. Van Rossum and F. L. Drake Jr, *Python reference manual*. Centrum voor Wiskunde en Informatica Amsterdam, 1995.

[19] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: https://www.tensorflow.org/

[20] NVIDIA, P. Vingelmann, and F. H. Fitzek, "Cuda, release: 10.2.89," 2020. [Online]. Available: https://developer.nvidia.com/cuda-toolkit

[21] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant, "Array programming with NumPy," *Nature*, vol. 585, no. 7825, pp. 357–362, Sep. 2020. [Online]. Available: https://doi.org/10.1038/s41586-020-2649-2

[22] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.

[23] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *Computer Vision – ECCV 2016*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. Cham: Springer International Publishing, 2016, pp. 21–37.

[24] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017.

[25] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2015.
[26] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *Computer Vision – ECCV 2014*, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds. Cham: Springer International Publishing, 2014, pp. 740–755.

Personalización de Recomendaciones en TripAdvisor Usando Deep Learning

Iñigo López-Riobóo Botana
Grupo LIDIA - CITIC - Universidad da Coruña.
Campus de Elviña, 15071 A Coruña, España.
Email: inigo.lopezrioboo.botana@udc.es

Amparo Alonso Betanzos
Grupo LIDIA - CITIC - Universidad da Coruña.
Campus de Elviña, 15071 A Coruña, España.
Email: amparo.alonso.betanzos@udc.es

Verónica Bolón Canedo
Grupo LIDIA - CITIC - Universidad da Coruña.
Campus de Elviña, 15071 A Coruña, España.
Email: veronica.bolon@udc.es

Antonio Bahamonde Rionda
MLGROUP - Universidad de Oviedo.
Calle San Francisco, 3, 33003 Oviedo, Asturias, España.
Email: abahamonde@uniovi.es

***Abstract*—Los sistemas de recomendación son una herramienta extendida en plataformas web de vídeo *streaming*, comercio electrónico, reserva de viajes u hoteles, entre otros, por lo que muchas empresas los ven con un valor de negocio potencial. Es común que se ofrezcan productos y servicios ajustándose a las preferencias de los usuarios para maximizar beneficios. En este trabajo, dado un ítem a recomendar por cualquier sistema de recomendación, nos centramos en cómo se presenta este resultado al usuario de forma personalizada. Proponemos el modelo *Text-Based and Personalised TripAdvisor Restaurant Recommendation* (TBP-TR2) en el contexto de la plataforma de críticas de restaurantes TripAdvisor. El objetivo es predecir para un usuario dado cual sería la mejor revisión textual del restaurante recomendado, de entre las disponibles para el mismo procedentes de otros usuarios. Este modelo hace uso de un modelo de lenguaje *transformer* pre-entrenado, *Bidirectional Encoder Representations from Transformers* (BERT) y lo adapta al propósito del trabajo, por medio de una red neuronal profunda a medida siguiendo una aproximación *feature-based* del modelo.**

I. Introducción

Los sistemas de recomendación se han convertido en una herramienta muy popular, cuyo impacto a la hora de generar valor en los procesos de negocio es innegable. Netflix, por ejemplo, reveló que "(...) el 75% de lo que la gente ve tiene origen en algún tipo de recomendación (...)" [1]. Amazon, por su parte, atribuye hasta el 35% de sus ingresos a los generados a través de su sistema *cross-selling* [2] (i.e., relacionado con la recomendación de “comprado junto habitualmente" y con la recomendación de “clientes que compraron x también compraron y"). Además, se sabe que el 93% de las compañías que contemplaron estrategias de recomendación experimentaron un incremento de sus ingresos en el ejercicio 2019 [3].

Sin embargo, no hay muchos estudios que trabajen en la personalización de dichas recomendaciones. Esto es, dado un sistema de recomendación arbitrario y un par (usuario, ítem), adaptar dicho ítem y presentarlo de forma diferente atendiendo al contexto del usuario. Partiendo de la idea de que conocemos la respuesta a **¿Qué recomendar?** (i.e., usuario e ítem), buscamos dar respuesta a **¿Cómo recomendarlo?**.

En nuestro caso, hemos elegido como aplicación la plataforma TripAdvisor, en un sector, el turístico, que representa el 12,4% del Producto Interior Bruto (PIB), y generó en 2019 el 12,9% del empleo sobre el total de la población activa, según [4]. Se puede trabajar en la personalización de ítems mediante recursos como imágenes o texto. Nosotros nos centraremos en una aproximación basada en texto, buscando las *reviews* adaptadas a cada usuario para presentar los restaurantes recomendados. Para ello, trabajaremos con técnicas de Procesado de Lenguaje Natural (PLN), incluyendo modelos de *deep learning* generadores de *word embeddings*.

Enmarcamos el contexto en una ciudad determinada, con su conjunto de usuarios, restaurantes e interacciones entre ellos (de forma similar a un grafo dirigido). Cuando un restaurante es recomendado, añadimos a dicha recomendación la *review* que mejor se ajuste a cada usuario, según estas interacciones. Idealmente, se correspondería a la *review* que habría escrito de haber visitado previamente el restaurante.

II. Trabajo Relacionado

El campo de los sistemas de recomendación ha sido ampliamente explorado. Existen multitud de propuestas para emparejar usuarios con ítem en aproximaciones colaborativas, demográficas, basadas en contenidos o híbridas [5], [6].

Centrándose en el sector hostelero y de viajes, particularmente sobre TripAdvisor, se suelen presentar un conjunto de puntos de interés a los usuarios. En **filtrado colaborativo**, se tratan los problemas de dispersión de la matriz usuarios-ítems y de *cold start* utilizando las puntuaciones para predecir nuevas [7]. Otras propuestas presentan recomendaciones basadas en similaridades de vecindarios [8]. ITAS [9] usa redes bayesianas para comparar usuarios frente a comunidades, generando puntos de interés. En **filtrado basado en contenidos**, hay aproximaciones que trabajan con listados de puntos de interés de forma personalizada, vinculados a series temporales [10]. *E-tourism* [11] y *MapMobyRek* [12] generan listados basados en las propiedades de los ítems que les gustan a los usuarios. En **filtrado demográfico**, hay propuestas

[13] que categorizan a los turistas por medio de diferentes clases (edad, sexo, etc) para recomendar puntos de interés. En **filtrado híbrido**, *Travigate* [14] recomienda puntos de interés usando una aproximación mixta de filtrado colaborativo y demográfico, por medio de un proceso de *clustering*. Otra aproximación híbrida [15] trabaja con *k-Nearest Neighbours* (k-NN) sobre filtrado colaborativo y de contenidos, además de árboles de decisión sobre filtrado demográfico. *Turist@* [16] es un sistema multi-agente centrado en filtrado colaborativo (*clustering* para determinar similaridades entre usuarios) y basado en contenidos (generar rankings de puntos de interés).

Sin embargo, **nuestro objetivo no es diseñar un sistema de recomendación** (que se asume pre-existente y arbitrario). **Nos centramos en la personalización de los ítems que se van a recomendar a los usuarios, atendiendo a sus preferencias**. Trabajando con imágenes, esta idea ya ha sido explorada en Netflix[1] y en otros ejemplos de TripAdvisor [17]. Sin embargo, hasta donde sabemos, se trata de un enfoque poco explorado en textos. Para ello usaremos *reviews* como principal soporte.

III. Materiales y métodos

En esta sección se detalla la extracción de datos de la plataforma TripAdvisor, así como la exposición de los conocimientos adquiridos sobre los mismos. Haremos también una breve mención a las técnicas de pre-procesado de texto utilizadas, así como al modelo BERT pre-entrenado del que se hace uso para generar *word embeddings*.

A. Conjuntos de datos. TripAdvisor.

Nuestro objetivo era seleccionar un grupo de ciudades conocidas con suficiente influencia angloparlante[2]. Se escogieron Madrid, Barcelona, Londres, Nueva Delhi, Nueva York y París[3]. La Tabla I resume las característica extraídas.

TABLA I
Características de los conjuntos de datos, sus tipos (Numérico, Fecha, Categórica y Texto) y descripciones.

Característica (N,F,C,T)	Descripción
parse_count (N)	Nº de review
user_id (C)	Identificador de usuario (UID_XXXXXXXXXX)
author (C)	Identificador de usuario (@autor)
restaurant_name (C)	Nombre del restaurante
rating_review (N)	Puntuación [1-5]
sample (C)	"Positive" [4-5] "Negative" [1-3]
review_id (C)	Identificador de la *review*
title_review (T)	Título de la *review*
review_preview (T)	Resumen de la *review*
review_full (T)	*Review* completa
url_review (T)	Url asociada a la *review*
date (F)	Fecha de publicación (día, mes, año)
city (C)	Ciudad del restaurante asociada a la *review*
url_restaurant (T)	Url del restaurante

[1]https://becominghuman.ai/how-netflix-uses-ai-and-machine-learning-a087614630fe

[2]Para trabajar con texto, optamos por el idioma inglés al ser el estándar más reconocido en tareas de PLN.

[3]Datasets disponibles en: https://udcgal-my.sharepoint.com/personal/inigo_lopezrioboo_botana_udc_es.

Para la recopilación de datos en TripAdvisor no existen APIs ni herramientas para investigación, por lo que se diseñó ad hoc un *web scraper*. Para ello se emplearon las tecnologías *Scrapy*[4] y *Selenium Web Driver*[5]. Ambas trabajan de forma colaborativa a la hora de generar los conjuntos de datos.

Atendiendo a la Tabla II, que resume la información más importante, se realizaron estudios sobre la calidad y distribución de los datos, de los se extraen las siguientes conclusiones:

1) Los usuarios **tienden a escribir *reviews* positivas**, como podemos ver al comparar el total de *reviews* en inglés respecto al subconjunto de positivas. Entendemos que las experiencias positivas motivan más a compartir una opinión, mientras que las negativas no producen ese efecto. Esto está directamente relacionado con la idea del *Customer Complaint Iceberg* [18]. Asumimos por lo tanto que los contextos de interacción negativos para cualquier restaurante están bastante incompletos.
2) Trabajando con **ciudades internacionales o con mucho turismo**, el hecho de considerar exclusivamente las *reviews* escritas en inglés no repercute negativamente sobre la cantidad de datos.
3) **La relación entre las *reviews* positivas escritas en inglés y usuarios únicos sigue una regla dos a uno**. Esto quiere decir que por cada dos *reviews* diferentes existe una cuenta de usuario. Esta distribución[6], ejemplificada para Londres en la Figura 1, demuestra que hay muchas cuentas inactivas (i.e., sin apenas interacciones con restaurantes), mientras que existen muy pocos usuarios activos. **Esta problemática es central en la propuesta de nuestro método**. Dependemos de esta información para personalizar las recomendaciones.



Fig. 1. Distribución del número de interacciones positivas por usuario para el conjunto de datos de Londres.

[4]https://scrapy.org/

[5]https://www.selenium.dev/documentation/en/webdriver/

[6]Para el resto de ciudades, la distribución es análoga. Disponibles en: https://udcgal-my.sharepoint.com/personal/inigo_lopezrioboo_botana_udc_es.

TABLA II
INFORMACIÓN RESPECTO A LA DISTRIBUCIÓN DE *reviews*, RESTAURANTES Y USUARIOS. **LOS USUARIOS Y RESTAURANTES ÚNICOS** ESTÁN CALCULADOS CONSIDERANDO EXCLUSIVAMENTE *reviews* POSITIVAS ESCRITAS EN INGLÉS (I.E., PUNTUACIÓN EN EL RANGO $[4, 5]$).

Conjunto	Nº*reviews* (todos los idiomas)	Tamaño	Nº*reviews* (inglés)	*Reviews* positivas (inglés)	*Reviews* negativas (inglés)	Usuarios únicos	Restaurantes únicos
Nueva Delhi	148,541	162.5 MB	148,303	123,095	25,208	59,796	5,147
Nueva York	517,904	559.1 MB	517,604	424,823	92,781	218,738	1,715
Madrid	216,565	198.9 MB	177,353	145,177	32,176	88,560	5,481
Londres	1,000,365	1.1 GB	998,939	833,453	165,485	441,821	1,827
Barcelona	426,785	428.3 MB	417,240	339,385	77,855	203,514	6,319
París	527,663	558.8 MB	510,084	401,589	108,495	219,340	11,004

Abordamos el problema como **clasificación multi-etiqueta**, donde cada muestra está asociada a una salida de N clases binarias (etiquetas). No son mutuamente exclusivas, por lo que se puede predecir más de una y de forma independiente para cada ejemplo a la entrada. Cada etiqueta corresponde a un usuario de salida del modelo (sobre el que personalizar la recomendación), explicado en mayor detalle en la Sección IV. Para generar el conjunto de datos etiquetado para los entrenamientos de **TBP-TR2**, seguimos el siguiente proceso:

- Cada ejemplo consiste de **una entrada** de texto preprocesada sobre la que el modelo BERT genera los *embeddings*. La **salida supervisada** se corresponde con el vector de longitud N con las etiquetas para los usuarios.
- **Solo consideramos *reviews* positivas**, en el rango $[4, 5]$[7].
- **Una etiqueta asignada con valor 1** indica **(1)** que el usuario es el autor de dicha *review* a la entrada o **(2)** la *review* pertenece a un restaurante sobre el que dicho usuario tiene una interacción positiva (i.e., por medio de otra *review* positiva diferente). Siguiendo la Figura 2, **nos basamos en la hipótesis de que si un usuario tiene una interacción positiva (autoría) con un restaurante, el resto de *reviews* positivas del restaurante también se asocian a una interacción positiva suya**.

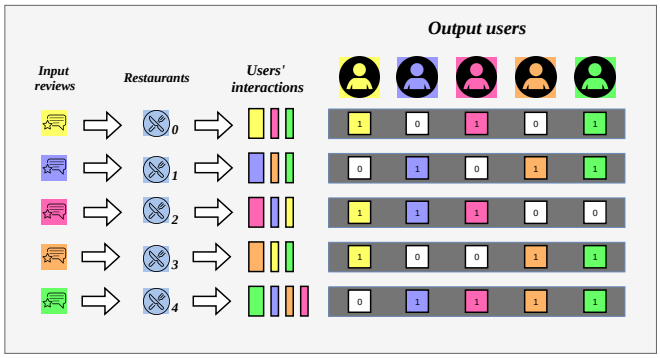


Fig. 2. Proceso de etiquetado de *reviews*. Cada texto está relacionado con un restaurante. La diagonal de la matriz define la autoría de los comentarios, mientras que otras posiciones positivas corresponden a otras interacciones positivas con el restaurante en cuestión.

[7]Basamos esta decisión en la dispersión de las interacciones positivas. El objetivo es mitigar el desbalanceo entre etiquetas positivas y negativas a lo largo de las muestras, por lo que el resto son filtradas.

- **Una etiqueta asignada con valor 0** indica "ausencia de información" en nuestro contexto (i.e., no se cumplen ninguna de las dos condiciones de etiquetado positivo).
- **Una *review* con un vector de salida con todas las etiquetas a 0** es descartado del conjunto, al no tener información alguna sobre interacciones positivas.

B. Métodos utilizados

Para trabajar con texto y utilizarlo como entrada a modelos de aprendizaje automático, necesitamos establecer una correspondencia entre el lenguaje natural y un espacio numérico real. Estas codificaciones se realizan de forma que secuencias que se encuentran cercanas en el espacio vectorial tengan significados similares. Antes de tratar estas representaciones, típicamente se aplica alguna técnica de preprocesado [19]:

- **Antes de generar *word embeddings***: ***lowercase*** para pasar todo el texto a minúsculas, sin considerar cambios de significado respecto a las mayúsculas. Además, **eliminación de la puntuación y caracteres especiales**.
- **Previo a evaluación (histogramas): *lemmatization***, para recuperar la raíz correcta de cada palabra y eliminar variantes. Por otro lado, **filtrado de *stopwords***, para descartar del vocabulario palabras de alta frecuencia que no aportan semántica al conjunto (determinantes, preposiciones, demostrativos, etc). Nuevamente, **eliminación de puntuación y caracteres especiales**.

Para generar *word embeddings*, optamos por una aproximación de redes neuronales profundas, *Bidirectional Encoder Representations from Transformers* (BERT) [20].

C. BERT

BERT [20] es un *transformer*[8]*-encoder* bidireccional y multicapa. En nuestra propuesta hacemos uso de la aproximación $\boldsymbol{Bert_{BASE}}$ pre-entrenada [21]. Siguiendo las recomendaciones de los autores para la aproximación *feature-based*, a la hora de generar los *embeddings* finales seguimos una estrategia de concatenación, juntando los estados de las últimas cuatro capas ocultas. Sabiendo que las secuencias a la entrada son de un máximo de 512 *tokens* y que para cada *token* se genera un vector de 768 posiciones, terminamos con un *embedding* de 3072 posiciones por *token* de la secuencia.

[8]Un *transformer* es una red neuronal profunda adaptada a PLN para procesamiento de datos secuenciales (e.g., texto), mediante una serie de mecanismos de atención sin importar el orden de la entrada.

BERT facilita la adaptación a numerosas tareas de PLN. Generalmente, actúa como bloque inicial de *mapping*, para generar los *embeddings* y posteriormente se configura el resto de la red para la tarea en cuestión (e.g., clasificación de texto, análisis de sentimientos, etc). Además, su flexibilidad permite trabajar con representaciones genéricas de la secuencia, así como con representaciones individualizadas de cada uno de los *tokens*. Gestiona secuencias formadas por una sola frase o pares de ellas, de forma transparente gracias a cómo está definida su entrada [20]. Además, modela el lenguaje de forma bidireccional, obteniendo la mayor cantidad de información del contexto. Por último, sobre el modelo BERT pre-entrenado, **siguiendo una aproximación *feature-based* [20], se puede reutilizar la totalidad de los pesos como las características principales para la definición de tu propia red neuronal profunda**. Esta es la aproximación seguida en nuestro caso. También existe la aproximación de *fine-tuning*, donde los pesos pre-entrenados se terminan de ajustar de forma fina y que puede ser mejor para determinadas tareas [22].

IV. Método propuesto

Partiendo de la problemática expuesta en la Sección III-A, dado un subconjunto de usuarios y un ítem (restaurante) predicho por un sistema de recomendación arbitrario, nuestro algoritmo **TBP-TR2** realizará la predicción de la mejor representación textual (*review*) para cada uno de ellos. La Figura 3 detalla el proceso.

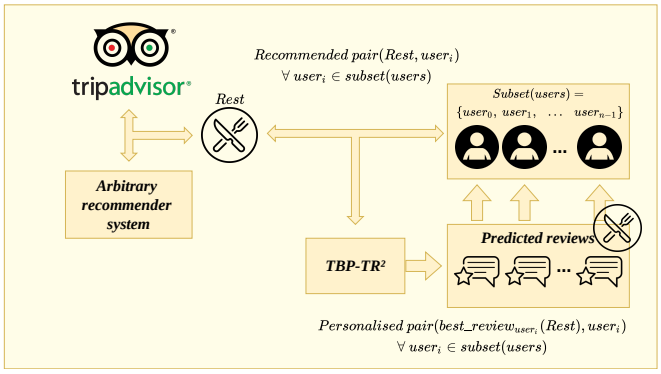


Fig. 3. Recomendaciones personalizadas usando *reviews* existentes del restaurante pre-seleccionado, dado un subconjunto de usuarios objetivo.

Dada una review positiva r del restaurante $REST$ y el subconjunto de usuarios U, queremos aprender la función:

$$f : r \rightsquigarrow \Big(valoración(u, REST) \geq 4 : u \in U\Big). \quad (1)$$

Por lo tanto, la probabilidad de que a un usuario $u \in U$ le guste cada *review*, $r \in reviews(REST)$ es:

$$\Pr(r|u) \quad \forall\ r \in reviews(REST) \wedge u \in U. \quad (2)$$

Queremos recuperar r^*, la *review* con el valor más alto de esa probabilidad, para cada usuario u:

$$r^* = \underset{r \in reviews(REST)}{\arg\max} \Pr(r|u). \quad (3)$$

Definimos una tarea de **clasificación multi-etiqueta para cada r a la entrada, correspondiendo al restaurante $REST$, sobre el subconjunto de usuarios U (etiquetas)**.

El modelo **TBP-TR2** está formando por dos bloques diferentes: **(1) Función de *mapping***, que se corresponde con el modelo $Bert_{BASE}$ pre-entrenado, explicado en la Sección III-C y detallado en la Figura 4. Por otro lado, **(2) Red adicional TBP-TR2**, para abordar el problema de clasificación multi-etiqueta expuesto, explicado en la Figura 5. Brevemente, se sigue una estrategia de concatenación de los embeddings BERT. Posteriormente se hace uso de una red bidireccional *Long Short-Term Memory* (LSTM) para gestionar las secuencias en series temporales y posteriormente aplicar un *pooling* (resumen) mediante medias y máximos de sus salidas. Se hace uso de una capa de *dropout* y regularización de pesos L2 para favorecer la capacidad de generalización y reducir el sobre-ajuste sobre el conjunto de entrenamiento. La capa de salida tiene activación sigmoide, para predecir probabilidades de forma independiente sobre cada usuario de salida.

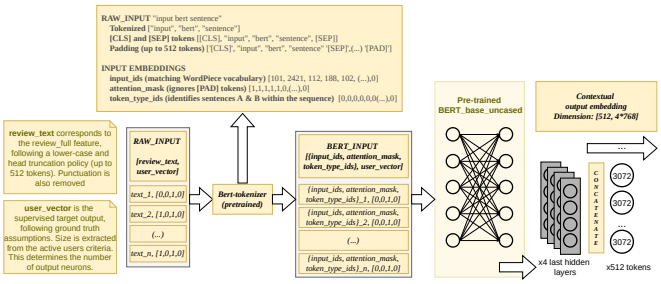


Fig. 4. Bloque inicial de *mapping*.

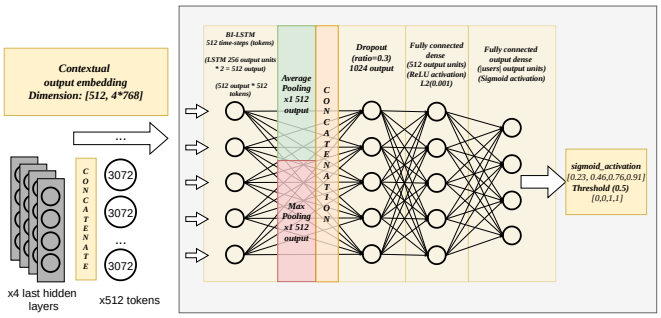


Fig. 5. Bloque personalizado **TBP-TR2**.

Para afrontar la dispersión de las interacciones comentada en la Sección III-A y facilitar los entrenamientos, definimos el **concepto de "usuarios activos".** Es el subconjunto de usuarios U definido previamente, cumpliendo con un umbral mínimo de interacciones en TripAdvisor. Es uno de los hiper-parámetros a configurar. Existe un compromiso entre el número de usuarios a la salida del modelo (tamaño de la capa de salida) y la cantidad de datos disponibles para entrenar.

V. Experimentación

Los experimentos constan de los siguientes pasos:

- **Para una ciudad determinada, se toman las *reviews* positivas escritas en inglés**. Se trunca la *review* desde el comienzo para un máximo de 512 tokens (el máximo permitido por BERT [20]). Se aplica preprocesado *lowercase* y se elimina puntuación.
- **Se descartan del conjunto de datos** las *reviews* asociadas al restaurante de evaluación. Es un subconjunto independiente del resto de particiones, utilizado para evaluar el modelo (histogramas de evaluación).
- **Se sigue el proceso de etiquetado** de la Sección III-A.
- **Se generan particiones pseudo-aleatorias** para entrenamiento (0,7), validación (0,15) y test (0,15). Se efectúan un total de 5 repeticiones de experimentos por conjunto de datos, con medias μ y desviaciones σ de las métricas en la partición de test.
- **Sobre la partición de entrenamiento,** se aplica la técnica de *oversampling* ML-ROS-20 [23], para compensar la falta de etiquetado positivo sobre la partición de datos.
- **Para cada experimento**, se vuelve a entrenar la red y se mezclan las particiones de forma pseudo-aleatoria.

En relación a los hiper-parámetros utilizados en la experimentación, se fijó un **tamaño de *batch*** de 16, con **optimizador** *Adam* y **ratio de aprendizaje** $3e{-}5$. Se establecieron un **máximo de 100 iteraciones**, definiendo un proceso de *early stop*, monitorizando con valor mínimo (Δ = 0,01) la función de pérdida *Binary Cross-Entropy* (BCE) en validación. Se fija margen de mejora para 5 iteraciones. Se restablecen los pesos de la mejor iteración (i.e., con el valor de BCE más bajo). **El umbral de "usuarios activos"** (Nºinteracciones mínimas) se corresponde con los valores: Barcelona (100), Londres (265), Madrid (70), Nueva Delhi (150), Nueva York (150) y Paris (123). **La ponderación del etiquetado positivo frente al negativo** se fija en 2. El objetivo es contrarrestar el desbalance positivos-negativos en el entrenamiento. Las etiquetas positivas tienen el doble de atención en la pérdida BCE, al ser minoritarias. Finalmente, se fija un **ratio de *dropout*** a 0,3 y **ratio de regularización L2** a 0,001.

En la Tabla III se resumen las principales métricas en test para el conjunto de experimentos[9]. **Analizando estos resultados**, los valores de *Area Under Curve* (AUC) tanto en el caso de *Precision-Recall* (PR) como de *Receiver Operating Characteristic* (ROC) muestran un buen ajuste del modelo. En general, comparando sensibilidad con especificidad, el modelo es menos sensible de lo deseado (especialmente **apreciable en el caso de París**, el ejemplo con mayor desbalance en el etiquetado). No obstante, **Madrid, Nueva York y Nueva Delhi** presentan muy buenos resultados en sensibilidad, manteniendo además valores altos de precisión (con las medidas de AUC-PR más altas del conjunto). Por eso fue importante considerar técnicas de *oversampling* y de diferente pesado para el etiquetado positivo, con el objetivo de suavizar la falta de interacciones positivas. Por último, la métrica de precisión balanceada (bACC), que tiene en cuenta ese desequilibrio en el etiquetado, resume un comportamiento general favorable.

Para tener una idea del funcionamiento de la personalización de las recomendaciones de un restaurante $REST$ sobre un usuario de salida u, comparamos dos histogramas diferentes[10]:

- **Contexto positivo general del restaurante** $REST$: Palabras más frecuentes de las *reviews* del restaurante de evaluación $REST$. Suelen asociarse a palabras genéricas que carecen de interés a la hora de determinar gustos o preferencias particulares.
- **Contexto positivo de un usuario de salida** u: Top 10 palabras más frecuentes del top 50 *reviews* predichas para el usuario u, dado el restaurante $REST$. **Excluyendo las palabras frecuentes y genéricas del histograma previo, podemos utilizar el resto de ellas como "contexto predicho" para representar sus preferencias**.

Para el caso de Barcelona, en el restaurante "Cera 23", el histograma general contempla como palabras más comunes "food", "restaurant", "good", "great" y "place", que carecen de importancia para las preferencias de usuario. Sin embargo, para el usuario de salida nº4 "John S", las palabras "blackberry" (top 2) y "mojito" (top 7) indican sus preferencias a este tipo de bebidas alcohólicas. *Reviews* que hablen de forma positiva sobre este aspecto del restaurante son las representaciones personalizadas más acertadas para este usuario. **En el caso de Nueva York**, en el restaurante "Boqueria", el histograma general contempla como palabras más comunes "tapas", "food", "good", "great" y "place", que nuevamente carecen de interés para los gustos específicos de usuarios. No obstante, al fijarse en el resultado para el usuario de salida nº0 "715autumnm", la palabra "sangria" (top 1) indica sus preferencias sobre este tipo de bebidas alcohólicas veraniegas, obteniendo nuevamente información clara sobre sus preferencias.

VI. Conclusiones y Trabajo Futuro

TBP-TR2 es un sistema que actúa como herramienta de soporte para la personalización de recomendaciones en la plataforma hotelera TripAdvisor, utilizando una aproximación basada en texto. Las *reviews* seleccionadas se corresponden con los resúmenes personalizados de los ítems (restaurantes). **La aplicabilidad de este sistema es directa, ya que puede utilizarse como segundo paso a cualquier sistema de recomendación pre-existente en la plataforma**.

Actualmente existen muchas aproximaciones para generar recomendaciones en TripAdvisor. Sin embargo, su personalización está poco explorada. La potencia y flexibilidad de BERT para generar *embeddings* sobre textos facilita mucho nuestra aproximación basada en *reviews*. Además, el ahorro en cómputo con modelos pre-entrenados es otro punto favorable a tener en cuenta. Es importante mencionar que la falta de datos estructurados y etiquetados, así como la escasez y dispersión de las interacciones en TripAdvisor, dificultaron el proceso de aprendizaje de **TBP-TR2**. Se hicieron necesarias ciertas

[9]Experimentos disponibles en: https://udcgal-my.sharepoint.com/personal/inigo_lopezrioboo_botana_udc_es.

[10]Se pueden consultar en la carpeta de resultados.

TABLA III
MÉTRICAS CALCULADAS SOBRE PARTICIÓN DE TEST.

Conjunto	**AUC-PR** ($\mu \pm \sigma$)	**AUC-ROC** ($\mu \pm \sigma$)	**Precisión** ($\mu \pm \sigma$)	**Sensibilidad (TPR)** ($\mu \pm \sigma$)	**Especificidad (TNR)** ($\mu \pm \sigma$)	**Precisión balanceada (bACC)** ($\mu \pm \sigma$)	**F-measure** ($\mu \pm \sigma$)
Barcelona	0.697 ± 0.006	0.856 ± 0.003	0.626 ± 0.003	0.643 ± 0.006	0.880 ± 0.003	0.744 ± 0.003	0.613 ± 0.003
Londres	0.723 ± 0.002	0.834 ± 0.001	0.602 ± 0.012	0.642 ± 0.016	0.824 ± 0.012	0.731 ± 0.005	0.619 ± 0.004
Madrid	0.772 ± 0.001	0.884 ± 0.001	0.694 ± 0.003	0.767 ± 0.005	0.836 ± 0.004	0.774 ± 0.005	0.689 ± 0.001
Nueva Delhi	0.799 ± 0.003	0.864 ± 0.003	0.697 ± 0.015	0.811 ± 0.014	0.736 ± 0.023	0.743 ± 0.006	0.708 ± 0.004
Nueva York	0.788 ± 0.005	0.889 ± 0.003	0.667 ± 0.008	0.732 ± 0.001	0.852 ± 0.005	0.780 ± 0.002	0.684 ± 0.005
París	0.635 ± 0.003	0.850 ± 0.001	0.640 ± 0.004	0.549 ± 0.004	0.929 ± 0.001	0.714 ± 0.003	0.552 ± 0.004

asunciones para definir un *ground truth* razonable. A pesar de tratarse de una tarea compleja, los resultados experimentales de la Sección V muestran un comportamiento general favorable del modelo, atendiendo a métricas como áreas bajo la curva (AUCs) o precisión balanceada. Otro escollo importante fue la búsqueda de métodos de evaluación fiables y robustos.

Como trabajo futuro, planteamos dos líneas: (1) La generación de nuevos y más diversos datasets sobre otras ciudades en TripAdvisor, para probar la efectividad de nuestro modelo. (2) La exploración de nuevos y mejores métodos para evaluación de los resultados. La fiabilidad y robustez de los procesos de evaluación son cruciales para poder validar el modelo propuesto.

AGRADECIMIENTOS

Este trabajo ha sido parcialmente financiado por el Gobierno de España (proyecto PID2019-109238GB-C22), por la Xunta de Galicia con las ayudas ED431C 2018/34 y ED431G 2019/01 y por los fondos ERDF de la Unión Europea. Especial agradecimiento también al Ministerio de Universidades por los fondos de la beca predoctoral FPU (FPU19/01457) y al CESGA (Centro de Supercomputación de Galicia) por el uso de la infraestructura de supercomputación Finisterrae II.

REFERENCIAS

[1] N. T. Blog, "Netflix Recommendations: Beyond the 5 stars (Part 1)," Apr. 2017. [Online]. Available: https://netflixtechblog.com/netflix-recommendations-beyond-the-5-stars-part-1-55838468f429

[2] C. Cohn, "A Beginner's Guide To Upselling And Cross-Selling." [Online]. Available: https://www.forbes.com/sites/chuckcohn/2015/05/15/a-beginners-guide-to-upselling-and-cross-selling/

[3] "Monetate Study Finds Brands Experiencing Highest Return on Personalization are Twice as Likely to Cite Customer Lifetime Value as Primary Goal," Mar. 2019. [Online]. Available: https://www.businesswire.com/news/home/20190313005156/en/Monetate-Study-Finds-Brands-Experiencing-Highest-Return-on-Personalization-are-Twice-as-Likely-to-Cite-Customer-Lifetime-Value-as-Primary-Goal

[4] "INEbase, Servicios, Hostelería y Turismo. Cuenta satélite del turismo de España, Últimos datos." [Online]. Available: https://www.ine.es/dyngs/INEbase/es/operacion.htm?c=estadistica_C&cid=1254736169169&menu=ultiDatos&idp=1254735576863

[5] I. Ryngksai and L. Chameikho, "Recommender Systems: Types of Filtering Techniques," *International journal of engineering research and technology*, vol. 3, 2014.

[6] F. Ricci, L. Rokach, and B. Shapira, *Recommender Systems Handbook*. Springer, 10 2010, vol. 1-35, pp. 1–35.

[7] N. E. E. Kyaw and T. T. Wai, "Inferring User Preferences Using Reviews for Rating Prediction," in *2019 International Conference on Advanced Information Technologies (ICAIT)*, 2019, pp. 194–199.

[8] R. Bidart, A. C. M. Pereira, J. M. Almeida, and A. Lacerda, "Where Should I Go? City Recommendation Based on User Communities," in *2014 9th Latin American Web Congress*, 2014, pp. 50–58.

[9] F.-M. Hsu, Y.-T. Lin, and T.-K. Ho, "Design and Implementation of an Intelligent Recommendation System for Tourist Attractions: The Integration of EBM Model, Bayesian Network and Google Maps," *Expert Systems with Applications*, vol. 39, no. 3, pp. 3257–3264, 2012.

[10] R. Mittal and V. Sinha, "A Personalized Time-bound Activity Recommendation System," in *2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC)*, 2017, pp. 1–7.

[11] L. Sebastia, I. Garcia, E. Onaindia, and C. Guzman, "E-Tourism: A Tourist Recommendation and Planning Application," in *2008 20th IEEE International Conference on Tools with Artificial Intelligence*, vol. 2, 2008, pp. 89–96.

[12] F. Ricci, Q. Nguyen, and O. Averjanova, "Exploiting a Map-Based Interface in Conversational Recommender Systems for Mobile Travelers," in *Tourism Informatics: Visual Travel Recommender Systems, Social Communities, and User Interface Design*, 2010.

[13] Y. Wang, S. C. Chan, and G. Ngai, "Applicability of Demographic Recommender System to Tourist Attractions: A Case Study on Trip Advisor," in *2012 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology*, vol. 3, 2012, pp. 97–101.

[14] V. Parikh, M. Keskar, D. Dharia, and P. Gotmare, "A Tourist Place Recommendation and Recognition System," in *2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT)*, 2018, pp. 218–222.

[15] M. E. B. H. Kbaier, H. Masri, and S. Krichen, "A Personalized Hybrid Tourism Recommender System," in *2017 IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA)*, 2017, pp. 244–250.

[16] M. Batet, A. Moreno, D. Sánchez, D. Isern, and A. Valls, "Turist@: Agent-based Personalised Recommendation of Tourist Activities," *Expert Systems with Applications*, vol. 39, no. 8, pp. 7319–7329, 2012.

[17] J. Díez, P. Pérez-Núñez, O. Luaces, B. Remeseiro, and A. Bahamonde, "Towards Explainable Personalized Recommendations by Learning from Users' Photos," *Information Sciences*, vol. 520, pp. 416–430, 2020.

[18] "What is the Customer Complaints Iceberg?" Sep. 2017. [Online]. Available: https://blog.v-comply.com/customer-complaints-iceberg/

[19] S. A. Ravi Lourdusamy, "A Survey on Text Pre-processing Techniques and Tools," *International Journal of Computer Sciences and Engineering*, vol. 06, pp. 148–157, 4 2018.

[20] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," *CoRR*, vol. abs/1810.04805, 2018.

[21] "BERT, transformers 4.4.2 documentation." [Online]. Available: https://huggingface.co/transformers/model_doc/bert.html?highlight=forsequenceclassification#tfbertmodel

[22] M. E. Peters, S. Ruder, and N. A. Smith, "To Tune or Not to Tune? Adapting Pretrained Representations to Diverse Tasks," in *Proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP-2019)*. Florence, Italy: Association for Computational Linguistics, Aug. 2019, pp. 7–14.

[23] F. Charte, A. J. Rivera, M. J. del Jesus, and F. Herrera, "Addressing Imbalance in Multilabel Classification: Measures and Random Resampling Algorithms," *Neurocomputing*, vol. 163, pp. 3–16, 2015, recent Advancements in Hybrid Artificial Intelligence Systems and its Application to Real-World Problems Progress in Intelligent Systems Mining Humanistic Data.

Sistema de Recomendación Transparente y Escrutable para Recomendaciones Personalizadas

Antonio Rodríguez-Revuelta
Centro de Inteligencia Artificial
Universidad de Oviedo
Gijón, España
UO11600@uniovi.es

Pablo Pérez-Núñez
Centro de Inteligencia Artificial
Universidad de Oviedo
Gijón, España
pabloperez@uniovi.es

Jorge Díez
Centro de Inteligencia Artificial
Universidad de Oviedo
Gijón, España
jdiez@uniovi.es

***Resumen*—Los sistemas de recomendación ayudan a los consumidores a encontrar productos que se ajustan a sus preferencias entre una gran cantidad de productos disponibles. Aunque las recomendaciones efectuadas por estos sistemas suelen ser acertadas, puede generar cierta desconfianza en el consumidor el hecho de no saber cómo ha sido generada dicha recomendación. En este artículo se muestra cómo construir un sistema transparente y escrutable, donde los consumidores comprenderán cómo se elaboran las recomendaciones y cómo podrían mejorarlas. Esto permitirá proporcionar buenas recomendaciones acompañadas de explicaciones comprensibles, como se muestra en los experimentos.**

***Index Terms*—Sistemas de Recomendación, Inteligencia Artificial Explicable, Multitarea**

I. Introducción

La inmensa cantidad de información a la que tienen acceso hoy en día los consumidores les permite tener la posibilidad de ver y comparar grandes cantidades de productos o contenidos antes de tomar una decisión sobre su compra o consumo. Esta situación, que a primera vista puede parecer beneficiosa para los consumidores, se convierte en una carga de trabajo adicional para los mismos, que pueden verse sobrepasados ante tal volumen de información. En este contexto surgen los Sistemas de Recomendación (SR) ([1], [2]) con el propósito de analizar los productos disponibles y los gustos de los consumidores para encontrar compatibilidades entre estos dos actores y así poder ofrecer al usuario un conjunto reducido de productos que se ajusten a sus preferencias.

Para evaluar los SR normalmente se atiende a medidas sobre la calidad de la predicción tales como la precisión, la exhaustividad o el error absoluto medio. Sin embargo, estas medidas no están teniendo en cuenta aspectos relacionados con la satisfacción del usuario. Es evidente que si una recomendación es acertada, esta debería producir satisfacción en el usuario pero, ¿por qué tenemos que fiarnos de la recomendación que nos hace el sistema? En este punto es donde entra en juego una disciplina que ha comenzado a cobrar importancia en los últimos años: la *Inteligencia Artificial Explicable*[3].

Este trabajo ha sido financiado por el proyecto PID2019-109238GB-C21 del Ministerio de Ciencia e Innovación de España. La colaboración de Pablo Pérez-Núñez ha sido financiada por el Gobierno del Principado de Asturias mediante el programa de becas predoctorales *Severo Ochoa* (ref. BP19-012)

Evidentemente, para que un SR tenga aceptación por parte de los usuarios debe ofrecer buenas recomendaciones pero debería, además, ofrecer algún tipo de explicación sobre la recomendación que está realizando ([4], [5]). De esta manera, el usuario podrá comprender cómo funciona el sistema (transparencia), detectar malas recomendaciones e intervenir en la mejora del modelo (escrutable), fiarse del sistema (confianza), ver aspectos del artículo que le pueden agradar (convincente), realizar buenas y rápidas elecciones (efectividad y eficiencia) y, por consiguiente, resultar satisfecho [6].

El volumen de datos con el que trabajan los SR ha propiciado que las técnicas que utilizan generen sistemas de tipo caja negra, donde las recomendaciones ofrecidas por el sistema no tienen una explicación directa que puedan entender los usuarios[7] y provoca que las explicaciones se conviertan en meras justificaciones de la recomendación [8].

En este artículo, vamos a mostrar una manera de adaptar un SR de tipo caja negra para que se convierta en caja blanca ofreciendo una total transparencia en sus predicciones y siendo escrutable por parte de los usuarios. Esto permitirá ofrecer explicaciones directas a los usuarios del por qué de la recomendación ofrecida y, además, se abre la posibilidad de que el usuario pueda modificar el sistema para adaptar las recomendaciones.

II. Trabajo Relacionado

En [9], los autores analizan diferentes tipos de explicaciones concluyendo que los SR de caja blanca ofrecen *explicaciones Cómo*, que son explicaciones en las que claramente se entienden los pasos que han llevado al SR a llegar a una determinada recomendación. Estas explicaciones ayudan a los usuarios a comprender cómo funciona el sistema y a incrementar su satisfacción [10]. Por el contrario, los SR de caja negra ofrecen *explicaciones Por qué*, explicaciones que son simplemente justificaciones del comportamiento del SR pero que no explican cómo funciona el sistema y que normalmente se basan en agregaciones de usuarios o productos cercanos.

Sobre esa misma idea se insiste en otros artículos donde se establece que *transparencia* es una declaración honesta de cómo se seleccionan las recomendaciones y de cómo funciona el sistema [11] mientras que *justificación* simplemente muestra

razones plausibles para la recomendación efectuada extraídas de diversas partes del algoritmo [8]. Además, los sistemas transparentes también son útiles para determinar cuándo un sistema tiene un cierto sesgo [7].

Cuando las explicaciones que da un SR proporcionan a los usuarios una forma directa y significativa de revisar su modelo estamos entonces ante un sistema *escrutable* [12]. Que un sistema ofrezca la posibilidad de ser escrutado por los usuarios permite perfeccionar los modelos ante situaciones en las que los intereses de un usuario vayan cambiando con el tiempo [13].

III. Método Propuesto

Consideremos un conjunto de usuarios $\mathcal{U}$, un conjunto de productos $\mathcal{P}$ y un conjunto $\mathcal{C}$ de características de los productos. Tenemos también registradas todas las relaciones en las que un usuario $u \in \mathcal{U}$ ha asignado una valoración v a un producto $p \in \mathcal{P}$ que tiene unas características $\mathbf{c} \in \mathcal{C}$. De esta manera, disponemos del conjunto de datos

$$\mathcal{D} = \{(u, [p, \mathbf{c}], v)\}\,. \tag{1}$$

Una manera habitual de abordar estos problemas de recomendación es mediante la factorización de matrices [14], donde se busca una función de utilidad basada, por ejemplo, en el producto escalar. En tal caso, se necesita una representación vectorial de u y p (que suelen venir identificados con un código) y para ello se utiliza frecuentemente una codificación *one-hot*, donde los vectores tienen como dimensión el número de elementos y todas sus posiciones serán 0 a excepción de la posición del código correspondiente al elemento que tendrá un 1, obteniéndose así las representaciones $\mathbf{u}$ y $\mathbf{p}$. Las características de los productos ya se tienen normalmente en un vector donde un 1 indica que posee una característica y un 0 que no la posee. De esta forma, una función de utilidad típica podría ser

$$utilidad = \langle \mathbf{W}\mathbf{u}, \mathbf{V}(\mathbf{p} \oplus \mathbf{c}) \rangle, \tag{2}$$

donde $\oplus$ denota la concatenación de los dos vectores. Esta función de utilidad puede sufrir una transformación posterior para adaptarse al tipo de valoración almacenada en el conjunto de datos (puntuación, me gusta / no me gusta, ...).

$\mathbf{W}\mathbf{u}$ y $\mathbf{V}(\mathbf{p} \oplus \mathbf{c})$ son la proyección de usuarios y productos, respectivamente, en un espacio de k variables latentes donde se organizan de tal manera que su producto escalar refleja la valoración o compatibilidad de usuarios con productos, obteniéndose buenos resultados normalmente. Sin embargo, el espacio en el que se proyectan los usuarios y los productos no permite que los SR así desarrollados sean transparentes, ya que las variables latentes no son útiles para que los usuarios comprendan cómo se elaboró la recomendación.

Una manera de solucionar esta falta de transparencia podría ser proyectando a usuarios y productos en un espacio donde cada dimensión pueda ser interpretada y comprendida por los humanos y, que a su vez, el producto escalar en dicho espacio siga produciendo valoraciones acertadas. Teniendo en cuenta que disponemos de las características de los productos, en este artículo proponemos utilizarlas no como parte de la entrada, sino como uno de los objetivos a aprender por el sistema. En la Figura 1, se muestra la arquitectura propuesta. Se transforma el código de usuarios y productos a vectores *one-hot*, que se proyectan en un espacio latente de dimensión k_l (utilizando una capa *densa* o completamente conectada) y se aplica la función de activación *ReLU*. Posteriormente se realiza una nueva proyección, pero esta vez en un espacio de dimensión k_c que debe coincidir con el número de características que se tienen de los productos y al cual se le aplica la función de activación *sigmoide* con el fin de que los valores estén acotados en el intervalo [0,1]. A partir de los usuarios y productos proyectados en este espacio, que llamamos de características, avanzamos en dos direcciones:

- por un lado, se multiplican escalarmente ambos vectores con el fin de obtener la valoración que el usuario u otorgaría al producto p, y,
- por otro lado, se comprueba que la representación obtenida para el producto se asemeje lo más posible a la descripción de sus características.

Al tratar de hacer que los productos se asemejen lo más posible al vector $\mathbf{c}$ que define sus características, estaremos consiguiendo que los ejes de ese espacio de características tengan sentido, ya que cada uno de esos ejes hará referencia a la probabilidad de que un producto posea esa característica o de que a un usuario le agrade dicha característica. Con estos vectores ya se podrían ofrecer recomendaciones acompañadas de *explicaciones Cómo* puesto que se obtiene un sistema transparente donde el usuario puede comprender cómo se elabora la recomendación. Además, aplicando el producto elemento a elemento de estos dos vectores (o producto de Hadamard) se podrá conocer qué características influyen más en la valoración y al poderse escrutar el modelo se posibilitaría la mejora del mismo.

Por tanto, se plantea un problema en el que se deben abordar dos tareas: tratar de predecir correctamente la valoración y, al mismo tiempo, las características de los productos. Durante el aprendizaje intervendrán entonces dos funciones de pérdida que deberán ponderarse adecuadamente para obtener la función de pérdida final:

$$\mathcal{L}([v, \mathbf{c}], [\hat{v}, \hat{\mathbf{c}}]) = \rho \times \mathcal{L}_v(v, \hat{v}) + (1 - \rho) \times \mathcal{L}_c(\mathbf{c}, \hat{\mathbf{c}}), \tag{3}$$

donde $\hat{v}$ y $\hat{\mathbf{c}}$ son las predicciones que hace el modelo y ρ es un hiperparámetro que varía en el rango $[0, 1]$ y que se utiliza para ponderar el peso de las dos funciones de pérdida.

IV. Resultados

En esta sección mostraremos cómo el método propuesto es capaz de generar un modelo de caja blanca que ofrecerá de manera simultánea una recomendación y la explicación de cómo se obtuvo tal recomendación.

Comenzaremos describiendo el conjunto de datos utilizado y viendo cómo adaptar la arquitectura a dicho conjunto. Después mostraremos que el sistema presentado obtiene resultados similares a los que obtienen otros sistemas (en cuanto al acierto en la valoración) y veremos la influencia del peso ρ en el

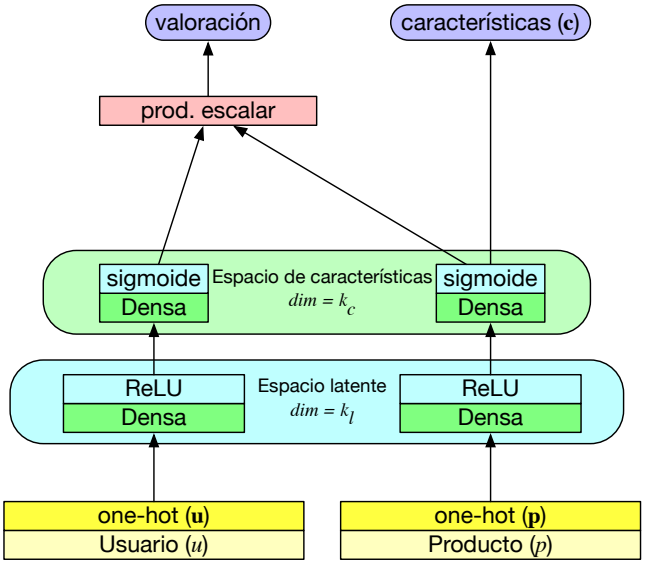


Figura 1. Arquitectura del sistema propuesto.

modelo. Posteriormente, analizaremos cómo es el espacio de características en el que se han proyectado tanto los productos como los usuarios para, finalmente, ver algunos ejemplos de explicaciones que pueden acompañar una recomendación.

IV-A. Conjunto de Datos Utilizado

Para nuestra experimentación hemos utilizado un conjunto de datos usado habitualmente como benchmark en tareas relacionadas con los sistemas de recomendación: *MovieLens 100K*[1]. Este conjunto cuenta con 943 usuarios donde cada uno de ellos ha valorado, al menos, 20 películas de entre las 1682 con las que cuenta el conjunto. En total, se encuentran registradas 100000 valoraciones numéricas entre 1 y 5, lo que supone que se desconoce el 93.7 % de las valoraciones posibles en la matriz entre usuarios y películas. Cada una de las películas tiene asignado, al menos, uno de los 18 géneros cinematográficos disponibles en el conjunto de datos (en concreto, la media de géneros por película es 1.83). También se dispone de una pequeña descripción de los usuarios (edad, sexo, ocupación y código postal) aunque en estos experimentos no se ha utilizado.

Un 10 % de las valoraciones se separó como conjunto de test, y de las 90000 valoraciones utilizadas como conjunto de entrenamiento, se separó, a su vez, otro 10 % como conjunto de validación para seleccionar los hiper-parámetros más adecuados.

IV-B. Detalles de Implementación

El sistema se implementó utilizando la librería TensorFlow [15], Adam como optimizador [16] y con regularización por el método de la *parada temprana*. Para adaptar la arquitectura propuesta en la Figura 1 al conjunto de datos utilizado en la experimentación, la dimensión del espacio de características k_c debe coincidir con el número de géneros, que en este conjunto es 18. El resto de hiperparámetros se seleccionaron observando los resultados obtenidos en el conjunto de validación: $k_l = 32$, como ratio de aprendizaje $1e-3$, tamaño de lote 512 y $\rho = 0.5$.

Como el conjunto utilizado contiene valoraciones numéricas, la función de pérdida $\mathcal{L}_v$ se corresponderá con el *error cuadrático medio* (problema de regresión) y, al especificarse los géneros de las películas con 0 o 1 dependiendo de si pertenecen o no al género, la función de pérdida $\mathcal{L}_c$ que se utilizará será la *entropía cruzada* adaptada a un problema multietiqueta.

IV-C. Calidad de la Predicción

Para ver si la arquitectura propuesta en este artículo puede ofrecer buenas recomendaciones es necesario comprobar su calidad a la hora de predecir la valoración que los usuarios otorgarían a las películas. Si somos capaces de predecir la valoración de manera correcta, entonces podremos recomendar a cada usuario la película a la que le otorgaría mayor valoración de entre las que no ha visto.

Hemos enfrentado nuestro sistema (al que llamaremos STE, *Sistema Transparente y Escrutable*) a otros modelos para comparar su rendimiento:

- FM_{oh} es un sistema tradicional que aplica factorización de matrices a partir de usuarios y películas codificados mediante vectores one-hot y que los proyecta en un espacio latente.
- FM_{oh+g} es una factorización de matrices en la que se incorporan los géneros en la codificación de la película tal como se indica en (2).
- GEN trata a los usuarios de la misma manera que se ve en la Figura 1 pero en la parte de las películas no se realiza ningún aprendizaje, considerándose que la representación en el espacio de características de las películas se corresponderá con los géneros que tienen asignados en el conjunto de datos. Por tanto, sólo podrán tener ceros y unos, lo que dotará al sistema de poca flexibilidad al hacer el producto escalar. En este caso, los usuarios se codifican mediante vectores one-hot y las películas mediante el vector de géneros a los que pertenecen.
- MED es el predictor *media* que utiliza la media de las valoraciones presentes en el conjunto de entrenamiento para asignar ese valor a todos los ejemplos del test. Utilizamos este sistema como *baseline* para identificar el valor del error por debajo del cual se puede concluir que los sistemas están aprendiendo. Este sistema no necesita codificar ni a los usuarios ni a las películas.

El objetivo de aprendizaje de nuestro algoritmo STE puede variarse utilizando el hiperparámetro ρ como se indica en (3). Para evaluar la calidad de la predicción hemos utilizado nuestro sistema con $\rho = 0.5$. En el siguiente epígrafe, veremos cómo afecta ρ al rendimiento de nuestro algoritmo.

En la Tabla I, se muestran los errores obtenidos en *Error Absoluto Medio* (EAM) por los sistemas descritos anterior-

[1]https://grouplens.org/datasets/movielens/100k/

Tabla I
ERROR ABSOLUTO MEDIO OBTENIDO POR LOS DIFERENTES SISTEMAS

Sistema	Representación		EAM
	Usuario	Película	
GEN	one-hot	géneros reales	1.6895
FM_{oh}	one-hot	one-hot	0.7111
FM_{oh+g}	one-hot	géneros reales & one-hot	**0.7063**
STE	one-hot	one-hot	0.7151
MED	–	–	0.9366

mente así como la manera en la que se representan tanto los usuarios como las películas. El baseline MED muestra que diciendo la valoración media en todas las predicciones, se obtienen un EAM de 0.936. Los sistemas FM_{oh}, FM_{oh+g} y STE son capaces de obtener un EAM inferior, con lo que estos sistemas sí están aprendiendo de los datos y proporcionando mejores predicciones. No ocurre así con el sistema GEN, que como sospechábamos tiene una arquitectura demasiado encorsetada y, el hecho de que las películas deban pertenecer completamente a un género sin permitir términos intermedios, hace que el producto escalar no tenga posibilidad de ofrecer buenas predicciones, llegando incluso a ser peor que el sistema utilizado como baseline.

Los algoritmos FM_{oh}, FM_{oh+g} y STE obtienen en este conjunto resultados muy similares y comparables a los que obtienen otros algoritmos en la literatura existente. Por tanto, esto muestra que el sistema STE diseñado para ser transparente y escrutable, también presenta un excelente comportamiento en cuanto a sus capacidades predictivas.

IV-D. Influencia de las Funciones de Pérdida

Una vez que ha quedado patente el buen rendimiento de la arquitectura propuesta en cuanto a la predicción de la valoración, mostramos en la Tabla II la variación en el rendimiento obtenido en las dos métricas analizadas (EAM y acierto) a medida que se varía el peso de las funciones de pérdida que optimiza el algoritmo propuesto: error cuadrático medio y entropía cruzada. En la ecuación (3) se muestra cómo interviene el ρ en la función de pérdida.

Cuando ρ es 1 y, por tanto, sólo se optimiza el error cuadrático medio vemos en la tabla que el EAM obtenido se sitúa en sus valores mínimos, en torno a 0.71. Si decrementamos el valor de ρ, el EAM se mantiene estable hasta un ρ de 0.5. Con valores de ρ menores el EAM empieza a incrementarse.

Para valores de ρ bajos se da más importancia en la optimización a la entropía cruzada del problema multietiqueta y, por tanto, el acierto sobre los géneros cinematográficos de las películas es bastante alto. Para valores de ρ mayores de 0.5, el acierto comienza a degradarse.

Los resultados de la tabla muestran que el sistema es bastante estable con valores de ρ que mantengan el equilibrio entre las dos funciones de pérdida optimizadas. Por esta razón, hemos decidido fijar el peso a 0.5 en estos experimentos.

La media de géneros que asigna nuestro sistema por película para ese $\rho = 0.5$ es 2.59, algo superior al 1.83 que tiene el

Tabla II
ERROR ABSOLUTO MEDIO OBTENIDO SOBRE LA VALORACIÓN Y ACIERTO OBTENIDO SOBRE LOS GÉNEROS DE LAS PELÍCULAS A MEDIDA QUE SE VARÍA EL PESO DE LAS FUNCIONES DE PÉRDIDA OPTIMIZADAS. $\rho = 1$ IMPLICA QUE SE PRESTA TODA LA ATENCIÓN A ACERTAR LA VALORACIÓN Y, POR EL CONTRARIO, $\rho = 0$ IMPLICA QUE SÓLO SE TIENE EN CUENTA ACERTAR LOS GÉNEROS DE LAS PELÍCULAS. TAMBIÉN SE MUESTRA EL NÚMERO MEDIO DE GÉNEROS PREDICHOS PARA LAS PELÍCULAS

ρ	EAM	Acierto	Media géneros
0	2.4774	0.9999	1.8235
0.1	0.8015	0.9998	1.8331
0.2	0.7477	0.9989	1.8776
0.3	0.7349	0.9955	1.9865
0.4	0.7259	0.9850	2.2607
0.5	0.7151	0.9644	2.5906
0.6	0.7143	0.9430	2.9181
0.7	0.7145	0.9111	3.3188
0.8	0.7076	0.8823	3.3792
0.9	0.7114	0.8265	3.5461
1	0.7121	0.5574	6.7321

conjunto de media, pero esto no es una mala noticia, ya que hay películas que aunque MovieLens no las haya etiquetado de un determinado género, tal vez, sí deberían haber sido consideradas pertenecientes a ese género (veremos un ejemplo más adelante).

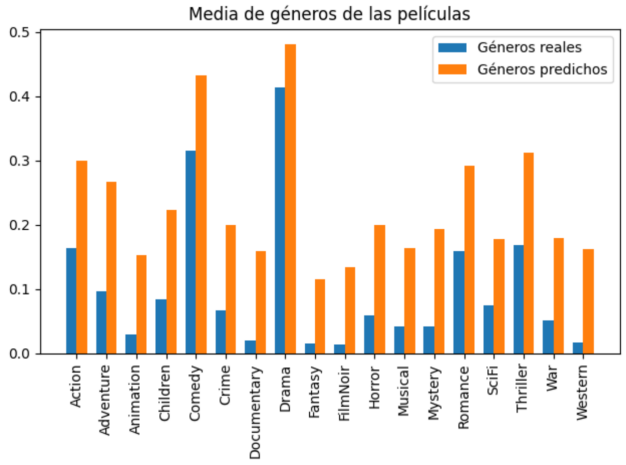


Figura 2. Valor medio de los géneros de las películas.

En la Figura 2 se muestra la media real de películas por género que contiene el conjunto de test y la media de géneros predichos para esas películas. Se puede apreciar que la correlación entre los géneros predichos y los reales es muy alta (concretamente 0.971) y que el sistema propuesto sobreestima los géneros. Esto se debe a que el sistema puede decidir que una película tiene, por ejemplo, un 0.25 del género *acción*, lo que indicaría que *tiene algo de acción pero no lo suficiente como para etiquetarla con ese género*.

IV-E. Explicación de la Recomendación

Como ya se comentó anteriormente, la arquitectura del sistema (Figura 1) se diseñó con la intención de proyectar tanto usuarios como películas en un espacio de características de los

productos, que en este caso se corresponde con los géneros de las películas, donde cada dimensión indica la afinidad que se tiene con cada uno de los 18 géneros cinematográficos disponibles (al utilizar una sigmoide como función de activación, el espacio se reduce al hiper-cubo que se forma entre 0 y 1 en esas 18 dimensiones). Para verificar esto, hemos calculado la intersección entre los 10 géneros con mayor influencia en cada usuario y los 10 géneros que más relación tienen con cada película y, posteriormente, hemos representado gráficamente la cardinalidad de estas intersecciones con respecto a la nota predicha por el sistema para todos los ejemplos de test. En la Figura 3 puede verse el gráfico resultante. Como esperábamos, se puede apreciar que tener menos elementos en la intersección está vinculado a valoraciones más bajas, mientras que más elementos está vinculado a notas más altas. Por tanto, analizando las componentes de las representaciones de usuarios y películas en ese espacio de géneros junto con el producto elemento a elemento de estos vectores, obtendremos una *explicación Cómo* (basada en grados de pertenencia a géneros) que permitirá a los usuarios entender cómo se ha calculado la predicción y cómo podría mejorarse en caso de ser necesario.

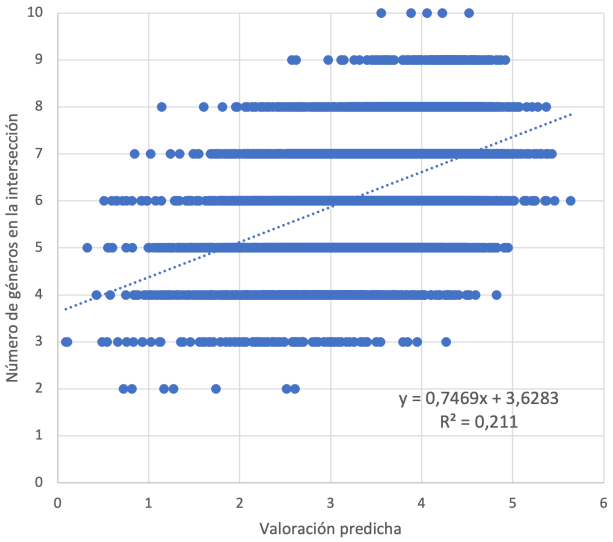


Figura 3. Valoración otorgada en los ejemplos de test (eje x) frente al número de intersecciones entre los 10 géneros preferidos por cada usuario y los 10 más relevantes de cada película (eje y).

En la Figura 4 se puede ver un ejemplo de recomendación acompañada de una explicación gráfica basada en los vectores que representan la proyección del usuario y la película en espacio de géneros. Cada radio de la figura muestra la afinidad del usuario y de la película con cada uno de los géneros y su producto da lugar a la valoración final. En la figura se aprecia el tipo de géneros que más interesan a este usuario, los géneros reales de la película *Fargo* y el grado de pertenencia a cada género predicho por el sistema para dicha película. Es interesante ver que esta película, está etiquetada por MovieLens con los géneros *Crimen*, *Drama* y *Thriller*, y que nuestro sistema le asigna, además, el género *Film Noir* (Cine Negro) con un grado de pertenencia de 0.506. Cabe destacar que si se accede a la ficha de esta película en la plataforma FilmAffinity[2] se puede ver que se le asigna el género *Neo Noir*, que es un género muy relacionado con *Film Noir*, con lo que vemos que nuestro sistema ha sabido detectar esa conexión y ha sido capaz de encontrar una mejor definición de la película en el espacio de géneros. En la figura también se muestra la nota media de las valoraciones otorgadas por el usuario y recibidas por la película junto con la nota que según el sistema ese usuario otorgaría a la película *Fargo*. Esta información es útil para que el usuario pueda situar la película en su propio baremo.

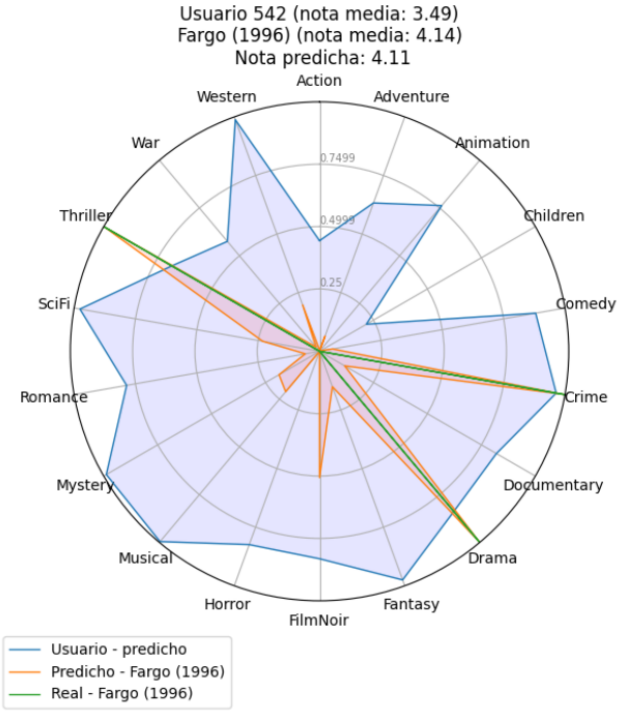


Figura 4. Ejemplo de recomendación.

Otra forma de aportar explicación en la recomendación es tratando de mostrar al usuario cómo se calcula la puntuación que predice el sistema. Teniendo en cuenta que la puntuación se obtiene realizando el producto escalar de la representación de un usuario y una película en el espacio de géneros que se ha aprendido, es muy sencillo indicar los géneros que más aportan a la nota predicha. Es de esperar que las películas que obtengan una mayor valoración para un determinado usuario estén vinculadas a los géneros que mayor afinidad tengan con el usuario en cuestión. El producto de Hadamard (elemento a elemento) del vector que representa al usuario por el vector que representa a la película en el espacio de géneros nos permitirá crear un ranking con los géneros más influyentes

[2]https://www.filmaffinity.com/es/film336111.html

en la valoración e incluso nos permitirá conocer con exactitud su influencia.

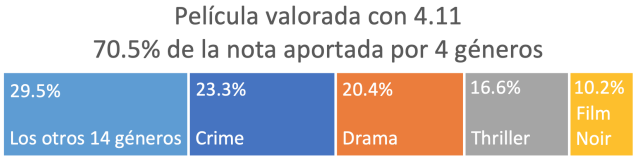


Figura 5. Explicación del aporte de cada género en una recomendación.

En la Figura 5 se muestra cómo podría ser una explicación basada en la aportación de cada género (inferidos por el modelo) a la nota predicha utilizando la misma recomendación mostrada en la Figura 4. Vemos los cuatro géneros que mayor valor obtienen tras calcular el producto de Hadamard y la importancia de cada uno de ellos para obtener la valoración predicha. En este caso se han utilizado para la explicación sólo los géneros que nuestro sistema asigna a la película, pero se podrían proponer otras explicaciones que mostrasen más información, como por ejemplo, mostrar todos los géneros que aportan más de un cierto umbral o mostrar todos los géneros que se necesitan para alcanzar una explicación del 90 % de la nota otorgada.

Cuando se tiene un SR totalmente transparente como el presentado en este artículo y el usuario comprende el mecanismo mediante el cual el sistema ofrece la recomendación, entonces, el usuario tiene la posibilidad de escrutar el modelo y de ser crítico con el funcionamiento del mismo. Es más, se le podría permitir retocar ligera y permanentemente su modelo (tal vez el usuario de la Figura 4, considere que *Animación* tiene demasiada importancia y que debería estar situada en torno a 0.5) o se podrían permitir ajustes temporales (ese mismo usuario recibe una visita y trata de adaptar sus gustos a los de sus invitados). Lo relevante es que el usuario conoce cómo es el mecanismo para las recomendaciones y sería capaz de intervenir para ajustarlo aún más a sus preferencias.

V. Conclusiones

En este artículo hemos mostrado una manera sencilla de crear un sistema de recomendación transparente y escrutable a partir de los datos que más habitualmente se tienen en este tipo de tareas. Las explicaciones que genera este sistema permiten al usuario analizar el perfil que ha generado el sistema partiendo de sus preferencias. Un perfil, que al estar expresado en afinidad con los géneros, es fácilmente comprensible. Además, al ser tan intuitivo, el usuario podría ser capaz de intervenir en el modelo para realizar modificaciones.

Todo esto se ha conseguido sin perder calidad en las recomendaciones realizadas, como se ha podido ver en la comparativa con otros sistemas, puesto que nuestro sistema plantea la solución como un aprendizaje multitarea.

Un último aspecto que es importante recalcar es que aunque en este artículo nos hemos centrado en proyectar en un espacio de características de los productos, se podría haber implementado un sistema equivalente proyectando en un espacio de características de usuarios (generando explicaciones sobre dichas características) o incluso combinando características de usuarios y productos.

Agradecimientos

Agradecemos a NVIDIA Corporation la donación de la GPU Titan Xp utilizada en esta investigación.

Referencias

[1] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, "Grouplens: An open architecture for collaborative filtering of netnews," in *Proceedings of the 1994 ACM conference on Computer supported cooperative work*, pp. 175–186, 1994.

[2] F. Ricci, L. Rokach, and B. Shapira, "Introduction to recommender systems handbook," in *Recommender systems handbook*, pp. 1–35, Springer, 2011.

[3] D. Monroe, "Ai, explain yourself," *Communications of the ACM*, vol. 61, no. 11, pp. 11–13, 2018.

[4] I. Nunes and D. Jannach, "A systematic review and taxonomy of explanations in decision support and recommender systems," *User Modeling and User-Adapted Interaction*, vol. 27, no. 3, pp. 393–444, 2017.

[5] Y. Zhang and X. Chen, "Explainable recommendation: A survey and new perspectives," *arXiv preprint arXiv:1804.11192*, 2018.

[6] N. Tintarev and J. Masthoff, "A survey of explanations in recommender systems," in *2007 IEEE 23rd international conference on data engineering workshop*, pp. 801–810, IEEE, 2007.

[7] E. Rader, K. Cotter, and J. Cho, "Explanations as mechanisms for supporting algorithmic transparency," in *Proceedings of the 2018 CHI conference on human factors in computing systems*, pp. 1–13, 2018.

[8] J. Vig, S. Sen, and J. Riedl, "Tagsplanations: explaining recommendations using tags," in *Proceedings of the 14th international conference on Intelligent user interfaces*, pp. 47–56, 2009.

[9] G. Friedrich and M. Zanker, "A taxonomy for generating explanations in recommender systems," *AI Magazine*, vol. 32, no. 3, pp. 90–98, 2011.

[10] F. Gedikli, D. Jannach, and M. Ge, "How should i explain? a comparison of different explanation types for recommender systems," *International Journal of Human-Computer Studies*, vol. 72, no. 4, pp. 367–382, 2014.

[11] N. Tintarev and J. Masthoff, "Evaluating the effectiveness of explanations for recommender systems," *User Modeling and User-Adapted Interaction*, vol. 22, no. 4-5, pp. 399–439, 2012.

[12] P. Pu, L. Chen, and R. Hu, "Evaluating recommender systems from the user's perspective: survey of the state of the art," *User Modeling and User-Adapted Interaction*, vol. 22, no. 4-5, pp. 317–355, 2012.

[13] K. Balog, F. Radlinski, and S. Arakelyan, "Transparent, scrutable and explainable user models for personalized recommendation," in *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 265–274, 2019.

[14] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, 2009.

[15] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D. G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, and X. Zheng, "Tensorflow: A system for large-scale machine learning," in *Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation, OSDI 2016*, 2016.

[16] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

Automatic Head Appearance Recognition for Individuals Affected by PIMD

Carmen Campomanes-Álvarez
Advanced Artificial Intelligence and Analytics Department
CTIC Technology Centre
Gijón, Spain
carmen.campomanes@ctic.es

B. Rosario Campomanes-Álvarez
Advanced Artificial Intelligence and Analytics Department
CTIC Technology Centre
Gijón, Spain
charo.campomanes@ctic.es

Pelayo Quirós
Advanced Artificial Intelligence and Analytics Department
CTIC Technology Centre
Gijón, Spain
pelayo.quiros@ctic.es

***Abstract*—Individuals affected by profound and multiple learning disabilities have severe restricted mobility and are subject to multiple sensory and intellectual impairments. They are unable to produce conventional behaviors to communicate their needs. The INSENSION project aims to develop an intelligent platform that allows these people to interact with digital applications for improving their quality of life. In this work, a head appearance recognition system, based on artificial intelligence, is proposed for the INSENSION platform. The goal is to identify the head poses of people affected by profound and multiple learning disabilities in response of an intention. The proposed system automatically recognizes diverse movements and positions of the head. Different algorithms have been trained and tested obtaining promising results, allowing to correctly recognize the poses required for persons affected by intellectual disabilities.**

***Index Terms*—artificial intelligence, machine learning, head poses recognition, head movements recognition, people with disabilities**

I. Introduction

People with Profound Intellectual and Multiple Disabilities (PIMD) present severe learning difficulties that seriously affect their skills to communicate their feelings and requirements, so their capacity to interact in common situations is severely diminished. They express their needs with non-symbolic behaviours, which are constituted by unconventional reactions like body movements or vocalizations. The interaction with these patients is done by understanding these emotional non-symbolic comportments and it requires a constant support by professional caregivers [1].

The goal of the INSENSION project is to develop an intelligent platform that enables persons with PIMD to increase their ability to self-determination with digital applications [1]. The platform is composed by a set of cameras and sensors that collect information from the environment of each person (Fig. 1). These data are analyzed by Artificial Intelligence (AI) algorithms, which final aim is to automatically associate a specific signal of a person to its meaning, providing these results through a digital service to the people in charge of them. The whole solution is composed by three stages: i) person identification [2], ii) behaviour pattern recognition, i.e., facial expressions [3], gestures, vocalization, and physiological parameters [4] and iii) interaction decision support service. Regarding the second stage, gesticulation is the use of motions of the body to communicate an intention or feeling. Current methods based on Computer Vision (CV) and AI for body gestures recognition includes tracking full or partial body motion in order to identify actions or human activities. However, gestures made by people with PIMD could have different meanings or communications attempts than those commonly known by society. For this reason, the analysis of the pose changes in people with PIMD implies an exhaustive study of each part of the body following the list of possible meaningful motions selected by pedagogical experts.

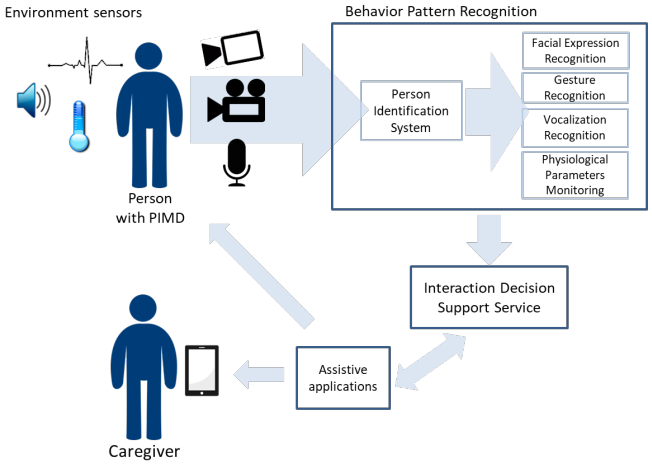


Fig. 1. General scheme of the personalized intelligent platform for enabling interaction with digital services to individuals with PIMD.

The aim of this work is to implement an automatic head appearance recognition system for people with PIMD within

the INSENSION platform. The general approach bases on the use of the body keypoints positions of the target individual in each frame of a video and AI-based techniques. Several classifiers and algorithms were developed in order to estimate the movement/pose of the head required to interpreter the actual meaning of a demand, protest or feelings that people with PIMD try to communicate.

This paper is structured as follows. Section II details a state-of-the-art study of head appearance recognition techniques. The proposed methodology is explained in section III. Section IV is devoted to describe the experimentation and results, and, finally, the conclusions of this work are outlined in section V.

II. Related Work

Regarding the automatic head movement detection, several studies have been developed by using head coordinates from Kinect sensors [5]. Other approaches based their research on detecting head movements by means of a methodology where Support Vector Machines (SVM) and Hidden Markov Models(HMM) models were trained to predict feedback nods and shakes in human-robot interactions [6]. Continuing with this later study, Morency et al. [7] tested a machine learning model to recognise head movements in video frames in a variety of datasets based on visual features obtained from tracked head velocities or eye gaze estimates extracted from video data.

On the other hand, Jongejan [8] proposed the OpenCV library to the detection of head movement from videos based on velocity and acceleration, in addition to customisable thresholds, for the automatic annotation of head movements using the ANVIL tool [9]. Furhermore, Jongejan et al. [10] analysed three visual movement features in order to train an SVM classifier of head movement. More recently, Paggio et al. [11] presented an approach to automatically detect and classify head movements in data from a corpus of video-recorded face-to-face conversations. A number of classifiers were trained with different combinations of visual, acoustic and word features and tested in a leave-one-out cross validation scenario.

However, in literature, any previous method does not identify special head movements in persons with PIMD, which is a crucial need for the INSENSION platform. Therefore, the present proposal is devoted to develop a new automatic head appearance recognition system for people with PIMD.

III. Proposed Methodology

The system consists of an automatic head appearance recognizer based on the positions of the body keypoints. The list of appearances to be recognized are: *shaking*, *nodding*, *raising*, *turning to side*, *leans to side* and *floppy*, which were established by pedagogical experts [13]. For this purpose, a set of classifiers has been developed to estimate the pose/movements. The system provides a value between 0 and 1, which indicates the degree of confidence of doing each action.

Let v the video stream provided by one camera (RGB data) at time t. A mapping between RGB data to body keypoints is shown as $v(t) \rightarrow p(t)$, $\forall t = 1, ..., T$, where *p(t)* represents the keypoints' position at time t and T represents the time length. *P(t)* consists of a list of 2D coordinates, namely $p\,(t) = \{(x\,(t)\,,\ y\,(t))\}_{i \in I,J}$, where i represents the keypoint index and I is the keypoint set defined by the pose detector mapping $I = \{0, .., 24\}$, and J the facial keypoint set $J = \{0, .., 69\}$. The location of each coordinate *(x, y)* are provided inside the processed frame according to a global coordinate system, where the origin of the axis is on the upper-left frame corner (Fig. 2).

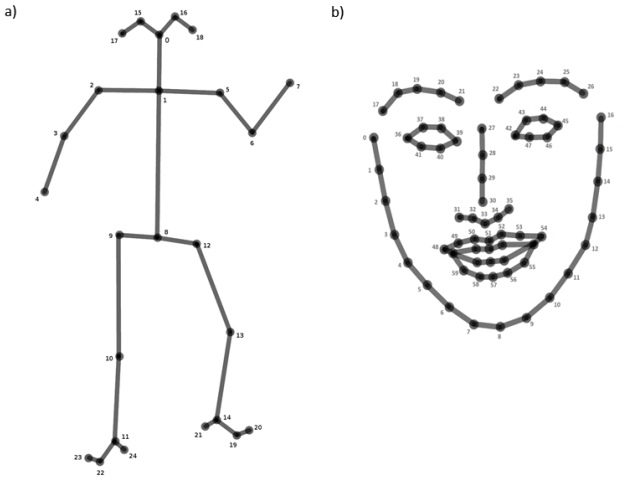


Fig. 2. 2D keypoints used for head appearance recognition. a) Body keypoints; b) Facial keypoints.

A. Head Movements Recognition

Some of the appearances of the head involve a movement, in particular, *shaking*, *nodding*, *raising*, and *turning to side*. From a technical point of view, a classifier able to distinguish between horizontal, vertical, non-directional, and no movement, is needed. Several classification models have been developed for classification using the following Machine Learning (ML) techniques: Naïve Bayes (NBs) [15], K-Nearest Neighbor (KNN) [16], Stochastic Gradient Descent (SGD) [17], Logistic Regression [18], Neural Networks (NNs) [19], and Random Forest [20].

1) Points Alignment: Firstly, the keypoints of the face are aligned with respect to the eyes axis in the center of the neck point. This approach allows measuring the movement of the head in the horizontal or vertical direction independent of the camera view. To do this, a canonical alignment of the keypoints based on a rotation is applied, in consequence, the head keypoints are rotated such that both eyes lie along the same y-coordinates.

2) Data Embedding: Secondly, a vector that contains the position of the aligned nose point for each frame of the sequence is stored. Using these values, the samples for the head movement classifier are created as follows:

$$S_{head\_mov} = \{\sigma(x_{nose}), \mu(x_{nose}), direction(x_{nose}), \sigma(y_{nose}), \mu(y_{nose}), direction(y_{nose})\}, \quad (1)$$

where $\sigma(x)$ and $\mu(x)$ are the standard deviation and the mean, respectively, and $direction(x)$ is defined by the following function:

$$direction(x) = \begin{cases} +1 & \text{if } x(T) - x(1) > 0 \\ -1 & \text{in other case} \end{cases}, \quad (2)$$

which provides the direction of the movement in the horizontal (x-coordinates) or vertical (y-coordinates) axis.

B. Head Pose Recognition

The other appearances of the head to be recognized require to know the pose of the head: *leans to side* and *floppy* head. The head pose estimation problem is often referred as a Perspective-n-Point problem (PNP) [14]. In this approach, the goal is to find the pose of the head with respect to the shoulders axis basing on the locations of 68 3D points on a model face and their corresponding 2D projections in the image. This method is based on the fact that a 3D object has only two kind of motions with respect to a camera: translation and rotation. Therefore, estimating the pose of a 3D object consists of finding the translation and the rotation vectors. In order to calculate the 3D pose of an object in an image three kinds of information are needed:

- 2D coordinates of the points: The 2D (x, y) locations of the 68 facial points in the image.
- 3D locations of the same points: 3D locations (x, y, z) of the 2D feature points are also needed. In this case, the 68 facial landmarks of a generic facial 3D model are used.
- Intrinsic parameters of the camera. In the PNP problem, it is needed to know the focal length of the camera, the optical center in the image and the radial distortion parameters. The device used in the INSENSION platform is the camera Logitech C920 which has a focal length of 3.67 mm (78º FOV) and no lens distortion. The optical center is approximated by the center of the image.

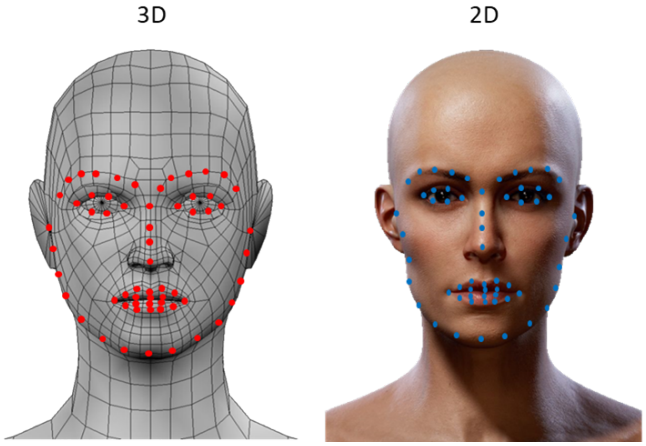


Fig. 3. Location in 3D and 2D of the selected facial landmarks for the head pose estimation.

The 3D coordinates of the various facial features shown above are in world coordinates. Once the rotation and translation (pose) is known, the 3D points can be transformed from world coordinates to 3D points in camera coordinates. The 3D points in camera coordinates can be projected onto the image plane (i.e. image coordinate system) using the intrinsic parameters of the camera (focal length, optical center, etc.).

Assume that the location (U, V, W) of a 3D point P in world coordinates is known. If the rotation R and the translation t regarding the world coordinates with respect to the camera coordinates are known, it could be feasible to calculate the location (X, Y, Z) of the point P in the camera coordinate system using the following equation:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = R \begin{bmatrix} U \\ V \\ W \end{bmatrix} + t \Rightarrow \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = [R|t] \begin{bmatrix} U \\ V \\ W \\ 1 \end{bmatrix}, \quad (3)$$

In the expanded form:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} r_{00} & r_{01} & r_{02} & t_x \\ r_{10} & r_{11} & r_{12} & t_y \\ r_{20} & r_{21} & r_{22} & t_z \end{bmatrix} \begin{bmatrix} U \\ V \\ W \\ 1 \end{bmatrix}, \quad (4)$$

The previous equation corresponds to a linear system of equations where the r_{ij} and (t_x, t_y, t_z) are unknowns. In the absence of radial distortion, the coordinates (x, y) of the point p in the image is given by:

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = s \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}, \quad (5)$$

Where f_x and f_y are the focal lengths in the x and y directions, and (c_x, c_y) is the optical center. Using the above equation, a (X, Y, Z) vector is obtained up to a scale s. This equation is solved using the method called Direct Linear Transform (DLT). The rotation (R) and the translation (t) vector obtained after solving the PNP problem, the head pose is known.

The rotation matrix R (6) can be converted into the three Eular angles that indicates the position of the head in terms of yaw, pitch and roll, as shown in Fig. 4a and (7), (8), and (9).

$$R = \begin{bmatrix} r_{00} & r_{01} & r_{02} \\ r_{10} & r_{11} & r_{12} \\ r_{20} & r_{21} & r_{22} \end{bmatrix} = R_x R_y R_z, \quad (6)$$

$$R_x(\psi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\psi) & -\sin(\psi) \\ 0 & \sin(\psi) & \cos(\psi) \end{bmatrix}, \quad (7)$$

$$R_y(\theta) = \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) \end{bmatrix}, \quad (8)$$

$$R_z(\phi) = \begin{bmatrix} \cos(\phi) & -\sin(\phi) & 0 \\ \sin(\phi) & \cos(\phi) & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (9)$$

The functions depicted in Fig. 4 determine the degree of having the head leans to side using the roll angle (ϕ) (b), the head turns to side using the yaw angle (θ) (c), and the head up, down or straight with the pitch angle (ψ) (d).

The appearance of the *floppy* head consists of leaning the head to a side, down or up, maintaining the same position for more than three seconds. To do this, the results of the no-movement class of the model explained in the previous section are considered for a sequence of frames equivalent to three seconds. The consecutive results of the head orientation and the values of the three angles (ϕ, θ, and ψ) are also stored for this time. Finally, these values are used to obtain the degree of having the *floppy* head using (10).

$$\mu_{head\_floppy} = \begin{cases} \overline{\mu}_{down} & \text{if } \overline{\mu}_{down} > 0.7 \text{ and } \sigma_{down} < 0.2 \\ \overline{\mu}_{leaning\_to\_side} & \text{if } \overline{\mu}_{leaning\_to\_side} > 0.7 \text{ and } \sigma_{leaning\_to\_side} < 0.2 \\ \overline{\mu}_{straight} & \text{in other case} \end{cases} \quad (10)$$

where $\overline{\mu}$ is the mean and σ is the standard deviation.

IV. Experiments and Results

All the experiments have been performed on an Intel CoreTM i7-5820K CPU 3.30 GHz x 12 with 31.3 GiB RAM and TITAN Xp Graphic card, running Ubuntu 16.04 LTS.

The keypoints are detected with OpenPose [21] and the public library scikit-learn [22] has been used for the implementation of the classification methods. A dataset has been created for the recognition of the head movements and poses. It is composed by recordings captured for this purpose by five different people from three different points of view (frontal, semi lateral-right and semi lateral-left) with static RGB cameras. A total of eight head actions have been recorded, including the neutral state which corresponds with no movements of the straight head. The details of the whole dataset are shown in Table I.

After the experiments, the model with the best trade-off achieved in the tests will be selected for each facial expression classification. This evaluation, involving exclusively the labelled dataset for each purpose, is conducted with the application of the weighted accuracy, weighted average precision, weighted average recall and weighted average F1-score metrics [23].

TABLE I
Details of the dataset of head movements and poses.

Head movement	Samples	Head movement	Samples
Turning to left side	63	Shaking	60
Turning to right side	62	Floppy	75
Raising	86	Leaning to side	60
Nodding	88	Neutral	60

TABLE II
Results of the experiments for the head movements.

Method	Accuracy	Precision	Recall	F1-score
Random Forest 50	0.89	0.89	0.89	0.89
K-Nearest Neighbour	0.87	0.88	0.87	0.87
Neural Network	0.80	0.84	0.80	0.76
SGD	0.80	0.68	0.80	0.73
Naïve Bayes	0.72	0.67	0.72	0.63
Logistic Regression	0.58	0.51	0.58	0.50

A. Head Movements Recognition

For the head movements recognition, the following actions are taken into account: *turning to left side*, *turning to right side*, *raising*, *nodding*, *shaking* and *no-movement*. The dataset has been split into a training and a testing dataset, in such a way that the samples for four people are used to train and the remaining one (a different person randomly selected) to test. The global results for each method are provided in Table II. Hence, the results show the weighted average of the six head movements for each metric.

The Random Forest method achieves the best global results for the head movements for the computed metrics (Table II). This method obtains 0.89 of accuracy and precision.

The confusion matrix of this model shows that some movements are more recognizable than others. In particular, *nodding* and *raising* have a rate of True Positives of 1.00, and *turning left* and *turning right* are also associated to high values, i.e., 0.87 and 0.86, respectively. Finally, *shaking* and *no movement* have 0.80 and 0.79, respectively, which is also acceptable. These results can be seen in Table III.

B. Head Pose Recognition

The poses *head leans to side* and *floppy* are modeled. The results are depicted in Table IV. The confusion matrices of such poses are represented in Tables V and VI, respectively.

The proposed method achieves promising results for the *floppy* head position with an accuracy equals to 0.86. Regarding the *head leans to side* pose, the accuracy slightly decreases

TABLE III
Confusion matrix of the head movements recognition.

	Predicted mov.					
Real mov.	***Neutral***	***Turn left***	***Turn right***	***Nod***	***Raise***	***Shake***
Neutral	0.79	0.07	0.07	0.00	0.07	0.00
Turn left	0.00	0.87	0.00	0.00	0.00	0.13
Turn right	0.00	0.00	0.86	0.00	0.00	0.14
Nod	0.00	0.00	0.00	1.00	0.00	0.00
Raise	0.00	0.00	0.00	0.00	1.00	0.00
Shake	0.10	0.00	0.00	0.10	0.00	0.80

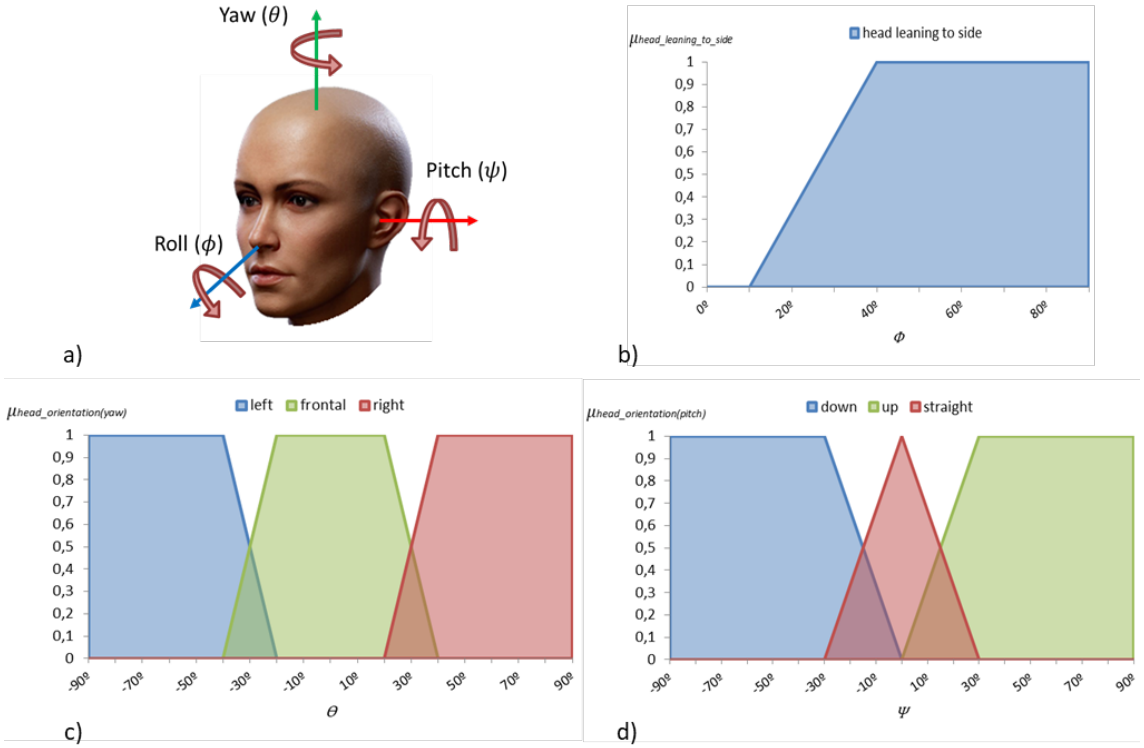


Fig. 4. a) Orientation of the head in terms of yaw, pitch and roll. b) Function to determine the degree of having the head leaning to side depending on the roll angle. c) Function to determine the degree of the three head yaw orientations: left, frontal and right. d) Function to determine the degree of the three head pitch orientations: straight, down and up.

TABLE IV
RESULTS OF THE EXPERIMENTS FOR THE HEAD POSES.

Pose	Accuracy	Precision	Recall	F1-score
Head leans to side	0.72	0.78	0.69	0.73
Floppy	0.86	0.88	0.84	0.86

TABLE V
CONFUSION MATRIX FOR THE POSE *head leans to side*.

	Predicted appearance	
Real appearance	***Head leans to side***	***Straight***
Head leans to side	0.79	0.21
Straight	0.35	0.65

with respect to the other head pose, but the approach performs well obtaining an accuracy of 0.72 and a precision of 0.78 (Table IV).

TABLE VI
CONFUSION MATRIX FOR THE POSE *floppy head*.

	Predicted appearance	
Real appearance	***Floppy***	***No floppy***
Floppy	0.88	0.12
No floppy	0.17	0.83

V. CONCLUSIONS

The INSENSION project aims to develop an intelligent platform that enables people with PIMD to improve the quality of their life by using digital applications.

In this work, a head appearance recognition system for this platform has been performed to automatically identify the head poses and movements in response to the intentions of people with PIMD. This component implies the development of a new approach, which analyzes particular appearance required for the application, defined by the pedagogical experts of the project. The proposed system differentiates between two kind of appearances: those which constitutes a movement of the head, i.e., *shaking*, *nodding*, *raising*, *turning to side*; and others that corresponds to a pose: *head leans to side* and *floppy* head. For the former, some ML algorithms have been trained and tested obtaining high-quality results (0.89 for all the metrics) with a Random Forest method. For the latter, the pose is obtained by solving a PnP problem, which provides the three angles that characterizes the position of the head regarding the shoulders' axis, i.e., yaw, pitch and roll. Using these values, a set of functions and equations are defined to obtain the degree of having the head *leans to side* or *floppy*. The performance of these methods arises an accuracy of 0.72, and a precision equals to 0.78 for the first case and, 0.86 and 0.88 for the second one, respectively.

In conclusion, a new methodology for the head appearance recognition for people with PIMD is proposed, to correctly discern head poses and movements that could be performed by these individuals for communicating their needs. The promising results obtained from the experiments show that these proposed techniques could be taken into account in further research that consider similar characteristics.

As further research within the INSENSION platform, new improvements to recognize patterns to others parts of the body, such as arms, hands, legs and foot, are needed.

Acknowledgment

This work has been conducted in the INSENSION project, funding from the European Union's 2020 research and innovation program under grant agreement no. 780819.

References

[1] Kosiedowski, M. et al.: On applying ambient intelligence to assist people with profound intellectual and multiple disabilities. Advances in Intelligent Systems and Computing, Cham: Springer International Publishing, 895–914 (2020).

[2] Campomanes-Alvarez, C. Campomanes-Alvarez, B. R., Quirós, P.: Person identification system in a platform for enabling interaction with individuals affected by profound and multiple learning disabilities. International Journal of Software Science and Computational Intelligence (IJSSCI), **12**(1), 30–46 (2020).

[3] Campomanes-Alvarez, C., Campomanes-Alvarez, B. R.: Automatic Facial Expression Recognition for the Interaction of Individuals with Multiple Disabilities. In Proceedings of the International Conference on Applied Artificial Intelligence (ICAPAI 2021). Digital conference, Halden, Norway (2021).

[4] Slapnicar, G., Dovgan, E., Cuk, P., Lustrek, M.: Contact-free monitoring of physiological parameters in people with profound intellectual and multiple disabilities. In Proceedings of the IEEE/CVF International Conference on Computer Vision Workshop (ICCVW). Seoul, Korea (2019).

[5] Wei, H., Scanlon, P., Li, Y., Monaghan, D. S., O'Connor, N. E.: Real-time head nod and shake detection for continuous human affect recognition. In 14th International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS), pp. 1–4. IEEE, Paris, France (2013).

[6] Morency, L. P., Sidner, C., Lee, C., and Darrell, T.: Contextual recognition of head gestures. In Proceedings of the 7th international conference on Multimodal interfaces (ICMI'05), pp. 18–24. Association for Computing Machinery Press, New York, USA (2005).

[7] Morency, L.P., Quattoni, A., and Darrell, T.: Latent-dynamic discriminative models for continuous gesture recognition. In 2007 IEEE conference on computer vision and pattern recognition, pp. 1–8. IEEE, Minneapolis, USA (2007).

[8] Jongejan, B.: Automatic annotation of head velocity and acceleration in Anvil. In Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12), pp. 201–208. European Language Resources Distribution Agency, Istanbul, Turkey (2012).

[9] Kipp, M.: Gesture Generation by Imitation: From Human Behavior to Computer Character Animation. Universal-Publishers, Boca Raton, Florida (2005)

[10] Jongejan, B., Paggio, P., Navarretta, C.: Classifying head movements in video-recorded conversations based on movement velocity, acceleration and jerk. In Proceedings of the 4th European and 7th Nordic Symposium on Multimodal Communication (MMSYM 2016), No. 141, pp. 10—17. Linköping University Electronic Press, Copenhagen (2017).

[11] Paggio, P., Jongejan, B., Agirrezabal, M., Navarretta, C.: Detecting head movements in video-recorded dyadic conversations. In Proceedings of the International Conference on Multimodal Interaction Adjunct - ICMI '18, pp. 1–6. AC Press. New York, USA (2018).

[12] Paggio, P., Agirrezabal, M., Jongejan, B., Navarretta, C.: Automatic detection of head movements in head-to-head conversations. In Proceedings of Language Resources and Evaluation Conference 2020 Workshop "People in language, vision and the mind" (ONION2020), pp. 15–21. Marseille, France (2020).

[13] M. Engelhardt, T. Krämer, M. Marzini, T. Sansour, and P. Zentel, "Communication assessment in people with PIMD. Evaluating the use of the INSENSION Questionnaire–Longform (InQL)," in Psychoeducational Assessment, Intervention and Rehabilitation, vol. 2, no 1, pp. 1–14, 2020.

[14] Lu, X. X.: A review of solutions for perspective-n-point problem in camera pose estimation. In Journal of Physics: Conference Series, vol. 1087, No. 5, pp. 052009. IOP Publishing (2018).

[15] Domingos, P., Pazzani, P. On the optimality of the simple Bayesian classifier under zero-loss: Machine Learning, **29**(2-3), 103-137 (1997).

[16] Hall, P., Park, B. U., Samworth, R. J.: Choice of neighbor order in nearest-neighbor classification. Annals of Statistics **36**(5), 2135–2152 (2008).

[17] Bottou, L.: Stochastic Learning. In: Bousquet O., von Luxburg U., Rätsch G. (eds) Advanced Lectures on Machine Learning. ML 2003. Lecture Notes in Computer Science, vol. 3176, pp. 146–168. Springer, Heidelberg (2003).

[18] McCulloch, C. E., Searle, S. R., Neuhaus, J. M.: Generalized, linear, and mixed models. 2nd ed. Wiley-Interscience, Newy York (2011)

[19] Picton, P.: Neural Networks. 2nd ed. Palgrave MacMillan, New York (2014)

[20] Ho, T. K.: The random subspace method for constructing decision forests. IEEE Transactions on Pattern Analysis and Machine Intelligence **20**(8), 832–844 (1998).

[21] Cao, Z., Hidalgo, G., Simon, T., Wei, S. E., Sheikh, Y.: OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields. IEEE transactions on pattern analysis and machine intelligence, **43**(1), pp. 172–186 (2019).

[22] Pedregosa, F., et al.: Scikit-learn: Machine learning in Python. the Journal of machine Learning research, **12**, pp. 2825–2830 (2011).

[23] Sammut, C., Webb, G. I. eds: Encyclopedia of machine learning. Springer, New York (2011)

Selección de características en una base de datos sobre el cáncer colorrectal

1st José A. Delgado-Osuna
Andalusian Research Institute in Data Science and Computacional Intelligence (DaSCI), Maimonides Biomedical Research Institute of Cordoba (IMIBIC), Universidad de Córdoba
Córdoba, España
jantonio.delgado.sspa@juntadeandalucia.es

2nd Daniel Ranchal-Parrado
Universidad de Córdoba
Córdoba, España
i62rapad@uco.es

3rd Carlos García-Martínez
Andalusian Research Institute in Data Science and Computacional Intelligence (DaSCI), Maimonides Biomedical Research Institute of Cordoba (IMIBIC), Universidad de Córdoba
Córdoba, España
cgarcia@uco.es

4th Sebastián Ventura
Andalusian Research Institute in Data Science and Computacional Intelligence (DaSCI), Maimonides Biomedical Research Institute of Cordoba (IMIBIC), Universidad de Córdoba
Córdoba, España
sventura@uco.es

***Resumen*—Se dispone de una base de datos de pacientes con cáncer colorrectal que se usó para extraer reglas de asociación interesantes mediante la inducción de árboles de clasificación. Sin embargo, la calidad de dichos árboles es muy limitada, con un porcentaje de acierto de apenas el 70 %. En el presente estudio analizamos la posibilidad de generar mejores árboles de clasificación, para dicha base de datos, mediante la aplicación previa de métodos de selección de características y de selección de instancias. Los resultados muestran una ligera ventaja de algunos de los métodos evolutivos utilizados sobre el resto de métodos y sobre la inducción del árbol sin selección de características.**

***Keywords*-Selección de características, selección de instancias, algoritmos evolutivos, Cancer colorrectal**

I. Introducción

El cáncer colorrectal es una patología de especial relevancia debido a su alta prevalencia y morbi-mortalidad. En España es la segunda causa de muerte por cáncer en el caso de los varones (por detrás del cáncer de pulmón) y la tercera en el caso de las mujeres (por detrás del cáncer de mama y de pulmón).

El cáncer colorrectal requiere una aproximación multidisciplinar en el que se ven implicados expertos sanitarios de diversas especialidades, quienes deben trabajar de una forma coordinada para reducir la tasa de complicaciones. Éstas se definen, en este estudio, como los eventos adversos posteriores a la intervención quirúrgica que afectan a la recuperación del paciente, estancia hospitalaria o reingreso.

Con el interés de mejorar la atención ofrecida a los pacientes de esta enfermedad, profesionales del Hospital Universitario Reina Sofía elaboraron una base de datos con 1516 pacientes y 126 atributos divididos en tres categorías:

1. Atributos quirúrgicos: se corresponden con 57 variables que proporcionan información quirúrgica (calidad de la resección, técnica quirúrgica usada, tipo de cirugía, etc.) sobre cada paciente en caso de que haya sido operado.
2. Atributos anatomopatológicos: se corresponden con 36 variables con información sobre los tejidos resecados al paciente en caso de que haya sido operado (tipo histológico, estadío, ganglios afectados, etc.).
3. Atributos sobre tratamiento oncológico: se corresponden con 33 variables que proporcionan información sobre el tratamiento quimioterápico que se ha aplicado, el seguimiento en consulta y el estado final del paciente indicando si tuvo recidivas o si superó la enfermedad (tratamiento administrado, suspensión de tratamiento, ocurrencia de recidivas locales o a distancia, etc.).

En [1], se propusieron métodos de extracción de reglas de asociación interesantes que se aplicaron sobre la base de datos anterior para la descripción de los casos con complicaciones. En el núcleo de dichos métodos se inducían árboles de clasificación, de los cuales se generaban las reglas iniciales. Sin embargo, las características de la base de datos, correlación baja entre las características y la predicción de complicaciones y la presencia de muchos valores perdidos, dificultan la obtención de árboles de clasificación con un elevado porcentaje de acierto. En el presente estudio nos planteamos analizar el efecto de aplicar diversos métodos de selección de características, algunos basados en algoritmos

Trabajo financiado por el Ministerio de Ciencia y Tecnología, TIN2017-83445-P

evolutivos, para la generación de árboles de clasificación más fiables para esta base de datos.

En la Sección II, describimos algunos aspectos de la base de datos. En la Sección III, presentamos los métodos de selección de características basados en algoritmos evolutivos utilizados. En la Sección IV, se muestran los resultados de los experimentos. Finalmente, las conclusiones se incluyen en la Sección V

II. La base de datos

La base de datos utilizada contiene información de 1516 pacientes caracterizados por 126 atributos. Algunos de dichos atributos son numéricos, como el número de días transcurridos entre la operación quirúrgica y la última revisión o fallecimiento del paciente, y otros son categóricos, como la técnica quirúrgica utilizada. Uno de esos atributos registra la aparición de complicaciones en el seguimiento al paciente, y es precisamente dicho atributo el que se desea relacionar con el resto. En dicha base de datos, 616 pacientes presentaron complicaciones y 900 no las presentaron.

De los 126 atributos, sólo el valor de 87 de ellos se establece con anterioridad a la aparición de complicaciones, por lo que serán estos los que realmente se utilicen en la generación de árboles de clasificación y, salvo que se indique lo contrario, serán los considerados en el resto de este trabajo. La base de datos presenta además un 12,36 % de valores perdidos, lo cual dificulta la generación de árboles de clasificación con un elevado porcentaje de acierto.

A continuación, se describen algunos de los atributos más relevantes de esta base de datos:

- KS: representa el tipo de cirugía (urgente o programada).
- ND: representa el número de días que el paciente ha permanecido ingresado en el hospital tras la intervención quirúrgica.
- SR: representa el estado de los anillos después de la sutura intestinal.
- NS: representa el número de cirugías a las que se ha sometido el paciente.
- SUR: representa al cirujano que realizó la operación quirúrgica.
- AS: representa al cirujano ayudante.
- ST: representa la técnica quirúrgica.
- DSD: es el número de días transcurridos entre la cirugía y el fallecimiento del paciente. Si el paciente aún continua vivo, representa el número de días transcurridos entre la cirugía y la última revisión.
- Endosponge: representa si el paciente ha sufrido una dehiscencia de la sutura que se ha tratado por vía transanal.
- QOR: evalúa la calidad de la resección en pacientes con cáncer de recto (en pacientes con cáncer de colón esta variable está en blanco).
- WPD: representa dónde fue diagnosticado el paciente.
- PSCI: representa el estado del paciente respecto al nivel de invasión del cáncer en la pared del intestino.
- HC: representa la clasificación histológica del tumor resecado.
- C: es la variable objetivo, y representa si una paciente ha tenido alguna complicación derivada de la cirugía.

III. Selección de características y de instancias con algoritmos evolutivos

Hemos utilizado dos algoritmos evolutivos para la selección de características de la base de datos, un modelo generacional y el algoritmo CHC [2], totalmente elitista. En ambos casos, hemos implementado los métodos en la librería de computación evolutiva deap [3], y se han parametrizado la función de evaluación de los individuos, y la posibilidad de usar selección de características en conjunción de selección de instancias:

- Algoritmo evolutivo generacional (AEG): Consiste en una población de 50 soluciones que se generan inicialmente de forma aleatoria. Se aplica el operador de selección por torneo con $k = 3$, el operador de cruce uniforme con una probabilidad del 50 % y el operador de mutación con una probabilidad del 20 %, el cual invierte el valor de cada gen con probabilidad 5 %. Cuando no se aplica ninguno de los operadores, se copian las soluciones padre. Al final de cada iteración, la población de descendientes reemplaza completamente a la población actual.
- CHC [2]: Consiste en una población de 20 soluciones que se generan inicialmente de forma aleatoria. Se forman parejas con las soluciones de la población mediante una selección aleatoria sin reemplazamiento, pero sólo se les permite generar nuevas soluciones a aquellas cuya diferencia supera un umbral dado. Dicho umbral se inicializa a un tercio del tamaño del problema y se reduce, el máximo entre 1 y el $2,5\,\%$ del tamaño del problema, cada vez que no se han producido parejas de soluciones con la distancia requerida. En caso de que el umbral llegue a ser igual o inferior a seis, se reinicializa la población con $N - 1$ soluciones aleatorias y se le añade la mejor solución hasta el momento.

En ambos casos se ha utilizado una codificación binaria en la que primero se codifican las características seleccionadas, si es el caso, y a continuación las instancias seleccionadas, igualmente si es el caso.

Para la evaluación de las soluciones se utiliza la métrica de calidad con la que se instancia el algoritmo, sobre un árbol de clasificación y una validación cruzada de 10 grupos, devolviendo bien la media o el menor valor, dependiendo de la instanciación del algoritmo. Se ha usado la biblioteca scikit-learn [4] tanto para la inducción del árbol de clasificación, como para el cálculo de las métricas de calidad. Para la generación del árbols, se ha ajustado la reducción mínima

de la impureza para realizar divisiones a 0.005. Las métricas de calidad utilizadas han sido el porcentaje de patrones bien clasificados (CCR), la métrica F1, la media de la precisión según la certeza de clasificación (MPCC), la media de la exhaustividad sobre cada clase (MECC), el coeficiente de similitud de Jaccard (Jac), uno menos el coeficiente de Brier (Br) y el área bajo la curva ROC (AUC).

Por último, ambos métodos se han ejecutado durante 10000 evaluaciones, devolviendo al final la mejor solución generada.

IV. Experimentos

Hemos realizado experimentos con los anteriores algoritmos evolutivos y los siguientes algoritmos heurísticos de la biblioteca scikit-learn [4][1], con sus valores por defecto: genérico univariable (GUS), selección por percentil (SP), selección de los k-mejores (SKBest), selección según ratio de falsos positivos (SFpr), selección según el error promediado por familias (SFwe), selección según el ratio de falsos descubiertos (SFdr), selección guiada por modelo de clasificación (SFromModel(Tree)), selección recursiva y guiada por modelo de clasificación (RFE(Tree)), selección recursiva con validación cruzada (RFECV(Tree)), selección según umbral de varianza (VarianceThreshold), selección secuencial voraz hacia adelante (SequentialFS_forward), y selección secuencial voraz hacia atrás (SequentialFS_backward). Es importante indicar que el uso de esta librería scikit-learn ha requerido que se utilice la codificación *one-hot* para las variables categóricas de la base de datos.

Para la evaluación de los modelos se han considerado las media, sobre una validación cruzada de 5 grupos y 2 repeticiones, del CCR, la métrica F1, el AUC de un árbol de clasificación entrenado con las características seleccionadas, y del tiempo de ejecución del método de selección.

La Tabla I muestra los resultados medios de los métodos. Para el caso de los modelos evolutivos se muestran únicamente los resultados de su mejor instanciación con o sin selección de características y de instancias, que son el generacional con la media del MPCC, el CHC con el mínimo del Br para el caso en el que se hace selección de instancias, y el CHC con la media del CCR para el caso en el que no se hace selección de instancias. Además, se ha considerado el árbol de clasificación entrenado con la base de datos original (No_sel_características). La Tabla I muestra también el resultado del test de wilcoxon de diferencias emparejadas entre el método con mejor media, en negrita y para cada métrica, y cada uno de los otros. Se utiliza el símbolo $\downarrow$ cuando el p-valor del test es inferior a 0.05, y $\approx$ en otro caso. Además, para simplificar el análisis, se ha añadido una columna en la que se cuenta el número de veces que el método ha obtenido resultados inferiores de acuerdo al criterio del test estadístico utilizado.

[1]https://scikit-learn.org/stable/modules/classes.html#module-sklearn.feature_selection

Podemos observar que los algoritmos evolutivos que realizan selección de instancias son los que han obtenido discretamente mejores resultados, ya que el resto de métodos consiguen resultados inferiores bien en F1 o en AUC. Por tanto, los métodos evolutivos no sólo están obteniendo clasificadores con elevadas tasas de acierto, sino que además están balanceando mejor las predicciones incorrectas entre las casos con y sin complicaciones. Por contra, se observa que dicha mejora se obtiene, por normal general, a costa de haber requerido mayores recursos computacionales. Sin embargo, los resultados intermedios, no presentados en este trabajo, sugieren que se podrían haber usado ejecuciones con menos evaluaciones.

Es particularmente curioso observar que el método que obtiene mejores resultados para las métricas F1 y AUC no utiliza selección de características, sino únicamente selección de instancias, por lo que la obtención de reglas de asociación propuesta en [1] también puede beneficiarse por este tipo de filtrado.

Por último, debemos indicar que somos conscientes de que los métodos heurísticos utilizados permiten un ajuste de sus parámetros que podrían haber ofrecido mejores resultados. Por ello, en otros estudios se podría invertir más esfuerzo en optimizarlos. En cualquier caso, el presente trabajo muestra que nos ha sido más sencillo obtener buenos resultados con los algoritmos evolutivos que ajustando los métodos heurísticos.

V. Conclusiones

En este estudio hemos analizado la posibilidad de generar árboles de clasificación más precisos, para una base de datos con información de pacientes con cáncer colorrectal, mediante la aplicación previa de métodos de selección de características y de selección de instancias. Hemos abordado el problema con dos algoritmos evolutivos y con métodos heurísticos presentes en la biblioteca scikit-learn [4]. Los resultados muestran una ligera ventaja para algunas de las instanciaciones de los métodos evolutivos.

Para futuros estudios nos planteamos analizar los componentes de los métodos evolutivos que están produciendo la observada ventaja sobre el resto de métodos de selección de características. En particular, deseamos analizar qué función o funciones de evaluación de los individuos resultan determinantes para producir los buenos resultados.

Referencias

[1] J. A. Delgado-Osuna, C. García-Martínez, J. Gómez-Barbadillo, and S. Ventura, "Heuristics for interesting class association rule mining a colorectal cancer database," *Information Processing & Management*, vol. 57, no. 3, p. 102207, 2020.

[2]Este método usó seis hilos en paralelo, mientras que el resto utilizó sólo uno

Cuadro I
RESULTADOS DE LOS MÉTODOS DE SELECCIÓN

Método	CCR	F1	AUC	T	# ↓
GUS	0.683↓	0.37↓	0.612↓	0s	3
SP	0.696≈	0.529≈	0.653↓	0s	1
SKBest	0.699≈	0.526≈	0.655↓	0s	1
SFpr	0.696≈	0.529≈	0.653↓	0s	1
SFwe	0.699≈	0.531≈	0.656↓	0s	1
SFdr	0.696≈	0.529≈	0.653↓	0s	1
SFromModel(Tree)	0.702≈	0.526↓	0.656≈	0.08s	1
RFE(Tree)	0.702≈	0.526↓	0.656≈	17.19s	1
RFECV(Tree)	0.702≈	0.526↓	0.656≈	2m	1
VarianceThreshold	0.702≈	0.526↓	0.656≈	0s	1
SequentialFS_forward	0.67↓	0.523↓	0.635↓	33m 53s[2]	3
SequentialFS_backward	0.622↓	0.537≈	0.609↓	39m 50s[2]	2
AEG(Sel_instancias,Sel_características)	0.703≈	0.544≈	0.661≈	48m	0
AEG(No_sel_instancias,Sel_características)	**0.714**	0.514↓	0.661≈	1h 17m	1
CHC(Sel_instancias,No_sel_características)	0.711≈	**0.56**	**0.671**	1h 23m	0
CHC(No_sel_instancias,Sel_características)	0.703≈	0.517↓	0.655↓	1h 2m	2
No_sel_características	0.702≈	0.526↓	0.656≈	0s	1

[2] L. Eshelman, *The CHC Adaptive Search Algorithm: How to Have Safe Search When Engaging in Nontraditional Genetic Recombination*, ser. Foundations of Genetic Algorithms. Morgan Kaufmann, 1991, vol. 1, pp. 265–283.

[3] F.-A. Fortin, F.-M. De Rainville, M.-A. Gardner, M. Parizeau, and C. Gagné, "DEAP: Evolutionary algorithms made easy," *Journal of Machine Learning Research*, vol. 13, pp. 2171–2175, jul 2012.

[4] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

Identification of outliers through classification based on evolution indices and their application to COVID-19

Yury A. Jiménez Agudelo
dept. Telecomunicaciones
Fundación Universitaria Compensar
Bogotá, Colombia
yajimeneza@ucompensar.edu.co

Victoria López
Quantitative Methods dept.
CUNEF University
Madrid, Spain
victoria.lopez@cunef.edu

Walter Chanava
dept. Ingeniería Informática
Universidad Nacional de Piura
Piura, Perú
walter.chanava@gmail.com

***Abstract*—Indices are transformations of the data useful to make the information interpretable. They are generally the result of aggregation or simplification, so their usefulness depends on the quantity and quality of the information contained. This study selects a set of useful indices to interpret the evolution of COVID-19 as input of the classification process. These indices have been defined from a dynamic flow network (that is also shown in this work) and some of them have been used by governments to inform the population and, above all, to take palliative measures where appropriate. In this work, we will see the indicators can also be used to classify regions or countries and especially to detect atypical elements through a ranking of the similarities between countries. To determine the similarity between the countries, the K-means classification algorithm has been applied over a predetermined period of time and with respect to 3 indices. The results demonstrate the usefulness of the ranking made from the results, especially for the detection of countries where the indicators present atypical values.**

***Index Terms*—Data analytics, classification, K-means algorithm, outliers, COVID-19 indices, flow dynamic model.**

I. Introduction

The SARS-CoV-2 virus was detected for the first time in December 2019 in the city of Wuhan (Hubei province, Popular Republic of China). It was declared by the OMS as a public health emergency of global importance on January 30, 2020, reaching the category of a pandemic on March 11, 2020 [1]. The COVID-19 disease (a consequence of the virus) was present in more than 180 countries in less than four months [2].

The spread occurred despite governmental measures taken worldwide trying to reduce the number of people infected. At the beginning of the pandemic, many countries around the world widely used the $R0$ number (reproductive number of infectious disease also called transmission rate) in determining political action. $R0$ number basically indicates how many people an infected person can give it to. However, the SARS-CoV-2 has a dynamic behaviour and by itself, the $R0$ index is an insufficient measure to assess the spread of infectious diseases [3], as it is also influenced by other aspects, such as the environment and the behaviour of the population.

Some studies show that there are cultural differences that affect the evolution of the disease in each country. The fatality is higher in the elderly and the infrastructures and coverage of the health services are very different in each country, variables that influence the SARS-CoV-2 virus evolution. As recent research papers have documented, the pandemic hits the poorest countries and regions the hardest [4]. Thus, the value of the $R0$ number may not accurately reflect the level of transmission in a given territory, and thus may lead to an error when determining the level of precaution.

Other variables that have been considered in determining political action, are lethality, accumulative incidence, intensive care unit occupation and a daily number of infected, deaths, recovery, among others. There are different models that establish a relation among some of the variables mentioned previously. Various studies have analyzed the different variables involved in the spread of the pandemic, many of them are inspired by the simplest compartment model SIR (Susceptible, Infected, Recovered), proposed by Kermack and Mc Kendrick in 1927. This model presumes that all the members of the population are going through the three states (or compartments): those who are susceptible (S), those who are infected (I), and eventually those who recovered (R). The number of people within these states is a function of time: S(t), I(t) and R(t) and the time unit is a day (24 h). Another model used for several research in this emergency ([5], [6], [7]) has been the SEIR model (Susceptible-Exposed-Infected-Recovered), this is an adaptation to the SIR model. Exposed denotes people that are exposed to the virus. In [8], the authors employ a compartmental model SEIR to describe the dynamics of the COVID-19 epidemic based on the epidemiological characteristics of individuals, the clinical progression of COVID-19, and social distancing measures. To measure the severity of the pandemic along the time in [9] a new dynamical model based on flow networks, inspired by the SIR model, is proposed. It presents a perspective of the evolution of the infected people among different states (infected, deaths and recovered). As a result, a danger index (IP) to measure the severity of the pandemic over time is proposed, this is a function of infected cases, number

of deaths and recovered cases.

The behaviour of the virus is influenced by the preventive actions carried out by each country, such as the measures dictated by their governments and the level of citizen compliance, country demography, the correct monitoring of security protocols, among others. This makes the spread and danger of the virus in each country particular, some countries have had a rapid expansion in the first weeks, while other countries had sustained growth [10] with low death rates.

This article makes a comparison of the rates of transmission, incidence and danger on the evolution of the COVID-19 in some countries of Latin America and Europe. As a result, a ranking that allows us to identify similarities between countries and among countries that had (in the evaluation period) a clearly different COVID-19 evolution (outliers) is presented.

This article is organized as follows. In addition to this introductory section, Section 2 explains the study's data sources and variables of interest. Section 3 details the mathematical model over which the indices are later defined within the same section and the procedure to find outliers. In Section 4 the results are shown and the indices are discussed. Finally, the conclusions are presented in the last section.

II. Data source and variables

This article makes a study of the data collected in open data sources. The study includes a set of variables of interest and three useful indices to evaluate the evolution of the pandemic. Some of these indices are used by governments to make decisions. This section also includes a new index whose justification comes from a dynamic model based on a flow network recently published in [9]. This section also includes the fundamentals and applications of this index.

A. Data source

Open data sources from governments have been used to get daily reported information about number of people: infected (INF/C), hospitalized (H), deceased (Lost/F), recovered (R) and in intensive care units (UC). Some important aspects of this variables have been taken into account, as described below.

- Start date of the pandemic: Latin American countries had their first official cases reported in March 2020, while the first cases were reported in European countries almost a month earlier. In this study, data are taken for each country counting from its own starting date, which allows us to make a comparison of the evolution during the initial period.
- Country size: The number of inhabitants per country must be taken into account in comparisons. The incidence rate (Inc) per million inhabitants has this value implicit. In addition, it must be taken into account in graphical representations for visualization to be useful.
- Data update frequency: Most of the data is published daily, but there are countries where data is not reported daily. In the latter cases, interpolation operations have been carried out on accumulated data to infer an estimate of the daily cases.

In addition to completing missing data, outliers have been eliminated, erroneous data detected and resolved, and individual smoothing of the variables of interest has been performed. The calculation of the indices ($R0$, Inc e IP) has been carried out with the smoothed variables and the results are shown graphically in the next section.

B. Variables

The variables data considered in this study are the absolute daily value, not accumulated. The selected variable are explained below.

- Infected or (C): People who have been tested positive for COVID-19, as well as people who the doctor has decided to classify as positive according to the symptoms presented, such as loss of smell and taste, constant feeling of fatigue, difficulty to breathe, among others.
- Hospitalized (H): People diagnosed through a medical test with COVID-19 and are sent or admitted to a hospital, due to their health condition.
- Diseased (F): People diagnosed through a medical test with COVID-19 and later die.
- Recovered (R): People who after being diagnosed with COVID-19 recover in a medical center or in their homes, this depends on the progress in their state of health.
- Intensive care units (UC): People who after being diagnosed with COVID-19 are admitted to the ICU for having a delicate state of health.

The data collected on these variables have been subjected to cleaning processes. On the one hand, missing data have been completed by interpolation in isolated cases. To reduce the noise produced by inadequate counts a smoothing has been carried out with simple moving averages. Next section shows the details.

III. Mathematical Model and Index Definition

This section describes the dynamic network flow model on which the variables and indices are defined in the study. First, the elements of the network are detailed and then the utility indices are defined.

A. Flow dynamic model

In flow-network theory, the transit from source nodes to sink nodes is key to preventing bottlenecks or collapses in the network. Figure 1 shows the flow network presented for the first time in [9]. The five nodes C, H, F, R, UC form a sub-graph in which nodes H (hospitalized) and IC (people in intensive care units) are internal nodes and therefore are the nodes susceptible to suffering bottleneck. Node C (infected) is a source node and nodes F (deceased) and R (recovered) are sink nodes.

Due to the characteristics of the problem, the aim is to minimize the flow that arrives through C and that that leaves

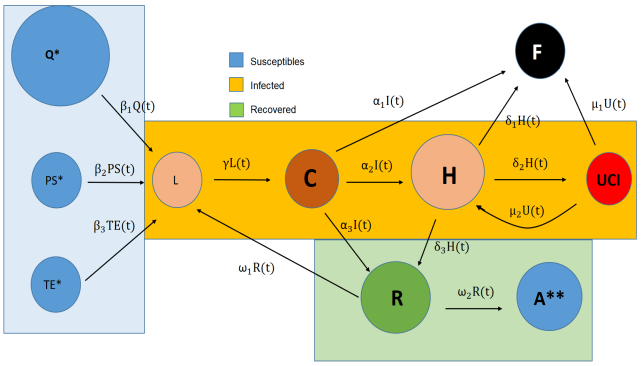


Fig. 1. Flow-Network model

through F and to maximize the flow that arrives at R, as defined in (1):

$$\forall k \in \{C, H, UC, R, F\},$$
$$\max_k f(C,k), \max_k f(k,R), \min_k f(k,F) \quad (1)$$

where $f(A,B) =$ daily flow from A to B.

B. Indices

Each country P has associated the variables C, H, R, F, UC described above. These variables are collected daily, however, they present some incorrect or unreported records. These cases have been solved by interpolation or moving average smoothing as described in (2) below.

$$X \in \{C, H, R, F, UC\}$$

Let x_i be the value of the variable X on the ith day in the country of study. The moving average of x_i is denoted by $mm(x_i)$ and its definition (for 7 days).

$$\forall i \geq 4, mm(x_i) = \frac{1}{7}\sum_{k=i-3}^{i+3} x_i \quad (2)$$

In this way, the values of the variables with a range of 7 days are 'smoothed'. The calculation of these means is a process of cleaning the data regarding count irregularities and they have been used by some governments to present their data publicly. The indices used are defined as follows:

- Transmission index (R_0): This index depends solely on the variable C (people infected daily) and in this study, it has been inferred with a 14-day perspective using equation (3). To measure the dangerousness of the virus, it must be compared to 1. A value less than 1 means that there is no danger.

$$\forall R_0^i = \frac{mm(c_i)}{mm(c_{i-14})}, \forall i \geq 15 \quad (3)$$

- Accumulated incidence per 100,000 inhabitants (Inc): This index also depends solely on the variable C, however, it is an index normalized by the number of inhabitants of the region or country of application, this is defined in (4). According to WHO recommendations, this index should be compared to 150. A value less than 150 means a medium-low level of danger. A level greater than 150 is a high level and greater than 250 is an extreme level. Figure 2 shows this index in a significant period in Spain. This figure also indicates the danger levels.

$$Inc^i = (\sum_{j=1}^{14} c_{i-j}) * 100000/P, \forall i \geq 15 \quad (4)$$

where P is the population of the corresponding country.
- Danger index (IP): This index is defined in [9] and is defined as shown in (5). The interest of this index is that it uses three variables C, F and R. These variables correspond to the source and sink nodes of the flow network model presented in the previous section. The index represents the equilibrium of the flow in the network, which is equivalent to controlling the bottlenecks in the internal nodes H, C, UC.

$$\forall IP^i = mm(c_i) + mm(f_i) - mm(r_i), \forall i \geq 1 \quad (5)$$



Fig. 2. Spanish incidence index (Source: Health Spanish Government)

The main difference between these indices is that the first ($R0$) and the last (Inc) only depend on the variable C (daily contagions) while the index IP also depends on the variable R (recoveries daily) and F (daily deaths). The $R0$ transmission index represents the virus's ability to spread; a 14-day window is usually used to calculate it as the ratio between infections at an instant t versus infections 14 days before. The cumulative incidence rate represents the number of cases per 100,000 inhabitants using the cumulative sum of infections in the last 14 days.

The interpretation of the indices is described below.

1) Interpretation of $R0$: This index only takes positive values and indicates the transmission ratio in reference to 14 days. This means that the number of infections on

that date is proportional to the number of infections 14 days before. $R0 > 1$ means the danger of transmission is greater than the value greater than $R0$. If $R0 \leq< 1$ it can be considered that the pandemic is under control, but to do so, this constraint must be maintained over time.

2) Interpretation of Inc: This index only takes positive values and is equal to the number of cases detected per 100,000 inhabitants accumulating the last 14 days, therefore, the higher it is, the greater the danger it detects. You can set a level above what is considered dangerous. For example, in Europe $Inc > 150$ is considered high risk, and $Inc > 250$ is extreme risk. $Inc < 150$ is a low-medium risk indicator.
3) Interpretation of IP: This index can take positive or negative values and must be compared with 0. If $IP > 0$ means danger and the higher the more danger. On the contrary $IP \leq 0$ means out of danger. This is the easiest index to interpret visually because it exhibits the greatest fluctuation over time. In those countries that do not publish data on recoveries, the IP index may still be useful since an average number of recovered RM can be estimated and the interpretation made by comparing with that number ($IP > RM$ means danger).

In the case of $R0$ and Inc, the graphical representation is shown normalized per million inhabitants to be able to compare several countries in the same graph. To visualize the results, the Rstudio ggplot2 library has been used, which allows making grids graph. In this way, the comparison of each variable for various countries is facilitated. Visualization is especially useful for comparing the first waves since in the observation period all the countries in the study had already had one or two waves. Figure 3 shows the evolution of these indices in 14 countries.

C. Outliers detection process

To detect significantly different countries, a novel procedure has been designed using the K-means algorithm [11] with k centers and a ranking created based on the number of connections or adjacency for each country. The procedure is explained below:

- Step 1. The data of the indicators $R0$, Inc and IP are filtered during a continuous period of time for a set of countries $\{p_1, ..., p_k\}$. As a result, three numerical matrices are obtained. In our study, we have chosen a continuous period of 90 days at the beginning of the pandemic for 14 countries: 12 from Latin America and two from Europe (to force the contrast).
- Step 2. The K-means algorithm is applied for each of the previous matrices with k centers, being k the number that fits better with the three indices. In our study that number is 6.
- Step 3. For each country p_i and for each resulting class, the adjacency number adj is calculated for each country. This number is the number of countries in the same class as the country in question.
- Step 4. The 3 adjacency numbers obtained for each indicator (R0, Inc and IP) are added. The result is the ranking (total adjacency) of each country, which is calculated using 6.

$$Ranking(p_i) = adj_i(R0) + adj_i(Inc) + adj_i(IP), \quad (6)$$

The following section shows the application results and the interpretation of these values.

IV. Results

In the first place, the values of the R0, Inc and IP indices have been studied for a set of 14 selected countries: 12 Latin American countries and two European ones, as mentioned above. Figure 3 graphically shows the evolution of the three indices in each country during the first 160 days. The start date depends on each country and corresponds to the date on which data begins to be published in the corresponding country. This same figure shows the differences in the indices for the same country. For example, Brazil and Chile are countries with similarities regarding the cumulative incidence (Inc) but different regarding the danger index (IP).

To test the quality of the indices, the K-means classification algorithm has been applied to the data of each indicator for a shorter period (90 days). The purpose of these classifications is to determine if the clusters obtained are similar in the three classifications for each index. The results are shown in Table I. In this table the number indicates the class to which each country belongs with respect to the index in each column. The number in parentheses indicates the number of items in the same class as the country corresponding to the row.

These values have been calculated using the process described in the previous section for locating outliers. For example, Bolivia has been classified in cluster 6 with respect to the $R0$ index and cluster 1 with respect to the other two indices. In cluster 6 of $R0$, in addition to Bolivia there are 3 other countries, which is indicated in parentheses. It follows, therefore, that Bolivia has similarities with 3 countries with respect to the $R0$ index. The values in the columns of the Inc and IP indexes are interpreted in a similar way. Bolivia has similarities with two countries in the classification regarding Inc and similarities with 4 other countries in the classification regarding IP. Therefore, the total ranking for Bolivia is $3 + 2 + 4 = 8$ connections detected. For the rest of the countries (rows) the interpretation is analogous. Finally, we observe that Peru and Spain are the countries with the lowest score, so they would be the outliers detected by the process. Brazil, with a score of 2 points, would be the next country on the list of atypical countries in the study period.

The result obtained can be compared with the results for each indicator, comparing the ranking with the countries that are classified in isolation: The $R0$ index returns Spain, the Netherlands and Brazil as isolated countries. The Inc index returns Chile and Peru as isolated. For its part, the IP index isolates Spain, Brazil, Paraguay and Peru.

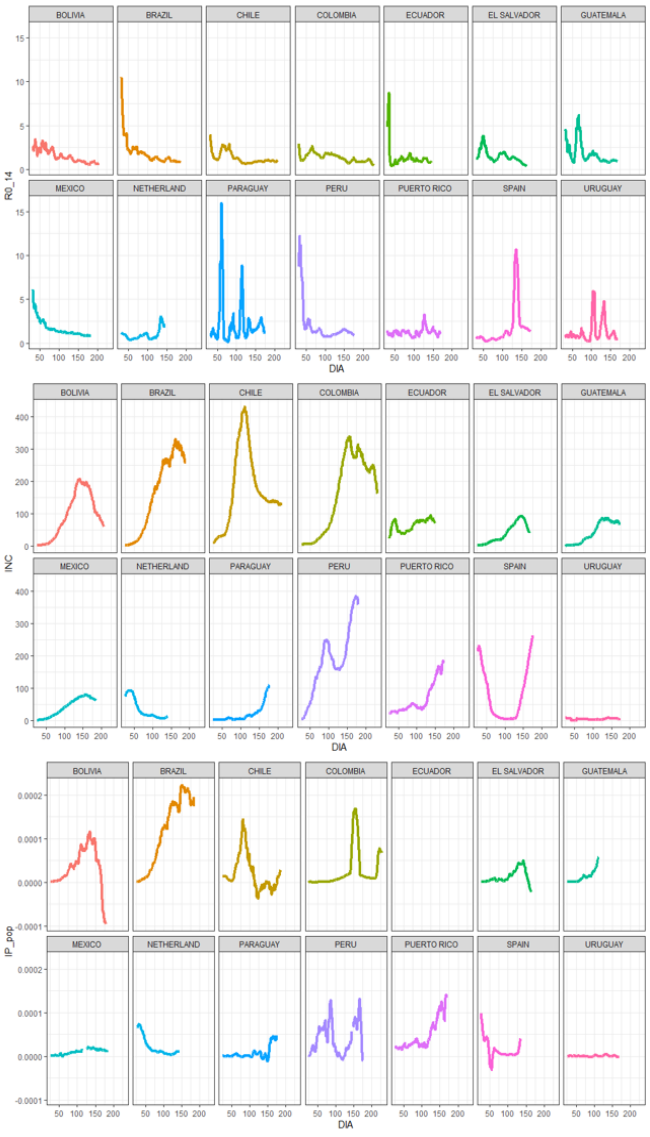


Fig. 3. Indices R0, Inc and IP along the first period of 160 days

V. Conclusion and Future Work

To assess the degree of danger of the pandemic, the different countries have used indices calculated from the data collected daily. However, these indices are not always consistent with each other. In this work, the result of its application over a period of time with real data has been analyzed and the results have been classified in order to detect not only groups of countries with similar values but also countries that are especially different with respect to said indicators. To measure the results, a ranking has been defined based on the number of connections found in the three classifications made using the K-means algorithm for each of the three indices in an equivalent period in all countries. The results show the countries that within the selected set show special significant differences. The process

TABLE I
K-means Classification and Ranking

COUNTRY	R0 (r)	Inc (r)	IP (r)	Adj_{total}
BOLIVIA	6 (3)	1 (2)	1 (4)	8
CHILE	3 (1)	4 (0)	1 (4)	5
ECUADOR	1 (5)	2 (4)	3 (4)	13
EL SALVADOR	1 (5)	2 (4)	3 (4)	13
GUATEMALA	6 (2)	2 (4)	1 (4)	10
MEXICO	6 (2)	2 (4)	1 (4)	10
PERU	3 (1)	6 (0)	6 (0)	1
PUERTO RICO	1 (5)	2 (4)	3 (4)	13
SPAIN	2 (0)	5 (1)	2 (0)	1
URUGUAY	1 (5)	3 (1)	3 (4)	10
NETHERLAND	4 (0)	5 (1)	3 (4)	5
COLOMBIA	1 (5)	1 (2)	1 (4)	11
BRAZIL	5 (0)	1 (2)	4 (0)	2
PARAGUAY	1 (5)	3 (1)	5 (0)	6

is useful to apply to various sets of countries and to specific time periods. As future work, the application to periods of time of 25 days around the peak of a wave is proposed for a group of countries within the same continent. The similarity and particularity of the countries may be of interest to interpret the palliative effect of the rules applied in each country.

Acknowledgment

This research has been supported by the project "RISE Women with disabilities In Social Engagement (RISEWISE)" under the Horizon 2020 programme (Grant Agreement: 690874) and also by the Spanish project VM-FAA (PID 2020-113192GB-I00).

References

[1] J. A. Sánchez-Duque, L. R. Arce-Villalobos, and A. J. Rodríguez-Morales, "Enfermedad por coronavirus 2019 (covid-19) en américa latina: papel de la atención primaria en la preparación y respuesta," *Atención Primaria*, vol. 52, no. 6, pp. 369–372, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0212656720301177

[2] A. Fanidi, X. Jouven, and B. Gaye, "Strategies to control COVID-19 and future pandemics in Africa and around the globe," *European Heart Journal*, vol. 41, no. 41, pp. 3973–3975, 04 2020. [Online]. Available: https://doi.org/10.1093/eurheartj/ehaa278

[3] B. Ridenhour, J. M. Kowalik, and D. K. Shay, "El número reproductivo básico (r0): consideraciones para su aplicación en la salud pública," *American Journal of Public Health*, vol. 108, no. S6, pp. S455–S465, 2018. [Online]. Available: https://doi.org/10.2105/AJPH.2013.301704s

[4] R. Verity, L. C. Okell, I. Dorigatti, P. Winskill, C. Whittaker, N. Imai, G. Cuomo-Dannenburg, Thompson, and et al., "Estimates of the severity of covid-19 disease," *medRxiv*, 2020. [Online]. Available: https://www.medrxiv.org/content/early/2020/03/13/2020.03.09.20033357

[5] S. Annas, M. Isbar Pratama, M. Rifandi, W. Sanusi, and S. Side, "Stability analysis and numerical simulation of seir model for pandemic covid-19 spread in indonesia," *Chaos, Solitons Fractals*, vol. 139, p. 110072, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0960077920304690

[6] L. López and X. Rodó, "A modified seir model to predict the covid-19 outbreak in spain and italy: simulating control scenarios and multi-scale epidemics," *medRxiv*, 2020. [Online]. Available: https://www.medrxiv.org/content/early/2020/04/16/2020.03.27.20045005

[7] P. Sun, K. Li, Z. Yang, and D. Du, "An seir model for assessment of covid-19 pandemic situation," in *Recent Featured Applications of Artificial Intelligence Methods. LSMS 2020 and ICSEE 2020 Workshops*. Singapore: Springer Singapore, 2020, pp. 498–510.

[8] C. Hou, J. Chen, Y. Zhou, L. Hua, J. Yuan, S. He, Y. Guo, S. Zhang, Q. Jia, C. Zhao, J. Zhang, G. Xu, and E. Jia, "The effectiveness of quarantine of wuhan city against the corona virus disease 2019 (covid-19): A well-mixed seir model analysis," *Journal of Medical Virology*, vol. 92, no. 7, pp. 841–848, 2020. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/jmv.25827

[9] V. López and M. Cukic, "A dynamical model of sars-cov-2 based on people flow networks," *Safety Science*, vol. 134, p. 105034, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925753520304318

[10] A. Mazumder, M. Arora, M. S. Sra, A. Gupta, P. Behera, M. Gupta, M. Agarwal, A. Rao, S. S. Mohanta, G. G. Parameswaran, and et al., "Geographical variation in case fatality rate and doubling time during the covid-19 pandemic," *Epidemiology and Infection*, vol. 148, p. e163, 2020.

[11] P. Flach, *Machine Learning: The Art and Science of Algorithms that Make Sense of Data*. Cambridge University Press, 2012.

Open data and experimental data source combination's impact on underground water analysis in Isla Margarita - Venezuela

Karina Gibert
Knowledge Engineering and Machine Learning Group at Intelligent Data Science and Artificial Intelligence Research Centre
Universitat Politecnica de Catalunya – Barcelona Tech
Barcelona, Spain
karina.gibert@upc.edu

Dante Conti
Knowledge Engineering and Machine Learning Group at Intelligent Data Science and Artificial Intelligence Research Centre
Universitat Politecnica de Catalunya – Barcelona Tech
Barcelona, Spain
dante.conti@upc.edu

Iñigo Arregui Pérez de Loza
University Research Institute for Sustainability Science and Technology
Universitat Politecnica de Catalunya – Barcelona Tech
Barcelona, Spain
inigo.arreguipdl@gmail.com

***Abstract*— Water is an increasingly scarce natural resource. An effective water monitoring, especially in areas where water shortage is structural and affected by climate changes requires of analysis tools which allow to document and analyze their endogenous behavior, so a better support on decision-making and strategies in water management processes is strengthened. This research presents a data mining approach based on clustering that aims to get profiles from water wells located in Pedro González-El Salado Sector aquifer (Margarita Island, Venezuela). The novelty of the research consists of enriching the experimental data with open data source information to increase the context knowledge and to facilitate the interpretation of the profiles/classes. Results can discriminate water quality from the groups within the clustering and open data information helps stablish a more robust context in terms of interpretability for each water well, and thus increases the available knowledge to face water scarcity and resources management in the case study.**

Keywords—data mining, hierarchical clustering, hydrology, open data sources, profiles interpretability

I. INTRODUCTION

Water is a first necessity resource for human population. As a matter of fact, United Nations has included both the access to the resource and the promotion of its sustainable management in the Sustainable Development's 6th Goal: ensure access to water and sanitation for all [1]. The increase in worldwide human population and climate change are determinant factors in the identification of water access problems in a coming future [2]. Therefore, different international institutions are working to address the use and treatment of water with the goal of guaranteeing the availability of high-quality water in a medium-large term.

The addition of subterranean water, and consequently aquifers, to the integral management of water resources holds a significant importance, given the low cost and low difficulties to access this type of water resource, which is crucial for the supply of one third of the world's population [3]. Operating aquifers is cheaper than managing water infrastructures. In addition, aquifers have a much greater water storage capacity than canals or reservoirs and thus, they are an important asset when addressing drought situations, especially in worldwide context in which climate change induced temperature increases will generate significant changes in precipitation and water distribution patterns, primarily in arid or semi-arid regions, where an intense water exploitation takes place [3]. This is the situation of Isla Margarita in Venezuela which endures a strong water scarcity situation according to the First Report on Water Problematic released by the National Assembly of the Bolivarian Republic of Venezuela [4] . The state has a daily water necessity of 225,5 million L, with a daily deficit of 121,8 million L. There are several water leakages, 980-1600 L/s in the water pipeline that connects the island with the continent. This situation has led to water supply cuts of 21 days in the island.

This research aims to address the characterization of Pedro González-El Salado aquifer which is one the most important water resource in the island by grouping its water wells and profiling them to increase the knowledge over the endogenous water resources of Isla Margarita with data related to water quality measures (experimental data). Thus, clustering techniques are applied to discriminate amongst groups the quality of the water. Open Data sources associated to the water wells are used to enrich the interpretation of the classes/profiles obtained. This is the novelty of the research which pretends to improve the context of the status of the aquifer by adding this new data as a tool to a better postprocessing of the results. Open data includes geographic, climatic, vegetation, urbanity, and economic situation in the region where the aquifer is located.

This paper will follow this structure: The next section contains the state of the art, then materials and methods are described to continue with the results, and finally conclusions & future works are listed.

II. STATE OF THE ART

Several efforts have been made in the Data Science field for analyzing water quality to improve water management. There are several significant research papers that show how a Data Mining approach can provide useful knowledge to address different situations. In 2011, researchers from Gazi University in Ankara, Turkey, analyzed Gökçekaya Reservoir's water's physical, chemical and biological traits along three years, to identify water quality variations depending on the season of the year [5]. A Principal Component Analysis (PCA) and Hierarchical Clustering techniques were performed over a set of variables composed by pH, Oxygen Dissolved in Water, Number of Dissolved Solids, Temperature, Electric Conductivity, Water Depth and Salinity; for determining seasonal variations caused by urban settlement processes and the own dynamics of the lake.

XXX-X-XXXX-XXXX-X/XX/$XX.00 ©20XX IEEE

A similar approach was followed in the analysis of the water quality in Hong Kong's Victoria Harbor [6]. In Reference [7] authors applied a rule-based clustering method, which incorporates expert's declarative knowledge to bias the hierarchical clustering method for the analysis of the performance of a Water Treatment Plant in Slovenia and getting more interpretable clusters. In addition, it was included a post-processing tool, called as Traffic Light Panel (TLP), to allow non-technical experts an easier interpretation of the results. Assigning one Traffic Light color to the values of each variable, for each class, non-statistician experts could access to an easier and more symbolic characterization of the results. This methodology was also applied to the analysis of a Water Treatment Plant in Girona, Spain [8].

Other types of clustering methods have been employed in water management field. In Reference [9] a Spectral Clustering approach was used to improve the urban water management by developing decision support services for leakage localization through a combination of clustering techniques and hydraulic simulation of leakage scenarios. Also, the combined use of graph and clustering methods have also been employed in the generation of direct potable water reuse (DPR) systems, considering both treatment plant scale, topography, and different household density [10].

III. MATERIAL AND METHODS

Data have been obtained from five different campaigns carried out by Venezuelan Ministry of Environment and Natural Resources, Nueva Esparta section, and corresponds to measurements of 36 water wells of Pedro González-El Salado aquifer during the years 2004, 2005, 2006, 2007 and 2008 adding up to 180 registers. For each register, dataset contains these variables: well identifier (Well-ID), electric conductivity (EC), static level (SL), pH, Universal Transverse Mercator East (UTMX) and North coordinates (UTMY), Longitude and Latitude. This dataset refers to experimental data.

Note that this research aims to remark the value-added by information coming from open-source databases (Open data) in order to provide a better context information associated to the water wells of the aquifer. Thus, experimental data of the aquifer were connected to external open data referring to: *Geographical variables* have been obtained to characterize the geographical condition of the terrain in which the water wells are located. Google Earth software has been employed to locate the wells over the map, drawing a line to the closest seashore point. This allowed to obtain an elevation profile for each well determining the Altitude, Distance to Sea and Slope of each water well. *Event Variables* have been considered to establish whether each water well does have a certain activity such as Lodging in their proximity. Every lodging establishment that appears in study has been located by means of SIG software QGIS, setting a proximity span of 500 m, to consider if a water well can be influence to a Lodging activity, and thus obtaining Lodging variable. *Territorial variables* describe the characteristics of the field in which the study takes place, giving information about the vegetation, lithology, and urbanism of the area. To get such information, two open data sources have been used. On the one hand, a Web Feature Service (WFS) provided by Venezuela Geographic Institute Simon Bolivar (IGVSB) has been used to obtain vectorial layers that contain information about Lithology and Vegetation variables, employing QGIS as the processing software. In addition, a non-digital map source has been employed to get urbanism information, gathering a map that covers the Vegetation and Land Use in Isla Margarita. *Climatic variables* have been obtained through Weather Online Ltd.'s webpage, as it has been impossible to get information from Venezuelan National Meteorology and Hydrology Institute (INAMEH). Medium Temperature variable has been obtained using such information source. Table I shows the description of the variables obtained from de water wells in the case study.

TABLE I. LIST OF VARIBLES IN THE CASE STUDY

Variable	Measure (Unit)	Description
Well-ID		Identification for each Water Well.
Year	Year	Year in which the measurement took place.
Electric Conductivity	mS/cm	Number of dissolved solids, depending on the capacity of the water to conduct electricity, commonly used for water quality measurement. MiliSiemens per cm.
Static Level	m	Level of subterranean water in wells with no pumping, referenced to sea water level.
pH		Level of acidity, indicating indirectly Hydrogen ion concentration.
Longitude	º	Angular distance between a point and the Greenwich Meridian, which takes 0º value. Occidental longitudes take negative values.
Latitude	º	Angular distance between a point and the Equator, which takes 0º value. Northern latitudes take positive values.
Altitude	m	Altitudes of each Water Well.
Slope	%	Slope of the terrain in which the Water Well is located.
Distance to Sea	km	Distance to the closer seashore point of each Water Well.
Lodging	-	Presence of Lodging economic activities in the proximity of each Water Well.
Vegetation & Land Use	-	Vegetation or Land Use of the terrain in which each Water Well is located.
Lithology	-	Lithology of the terrain in which each Water Well is located.
Medium Temperature	ºC	Medium temperature for every year in which measures took place.

Data were preprocessed and audited, and then, a data mining engine was performed applying clustering techniques. This is to group similar registers in homogenous groups, that differ significantly from other groups. The existence of different grouping algorithms requires to decide in advance which grouping method better adapts to the data. Since the optimal number of clusters is unknown, a connectivity-based grouping, such as the Hierarchical Clustering Method has been chosen. Two clustering experiments were carried out: (H1) Clustering with experimental dataset only and (H2) Clustering with the

whole dataset including the open dataset. Classes (profiles) obtained from H1 y H2 are then postprocessed. Once the characterization of each class has been established, a geographical visualization of the classes is built by using a Geographical Information System (GIS) software called QGIS.

IV. RESULTS

As it was mentioned before, clustering analysis has been performed through the application of an agglomerative grouping, such as the Hierarchical Clustering Method, computing distances with Gower Distance, and using Ward Method as the linking criteria. For H1, six profiles were identified (according to Calinski-Harabasz index): **Class 1** shows a medium level of Electric Conductivity, median of 2.30 mS/cm, while the Static Level of the wells are low, median of 4.95 m. The wells included in this class show low values for Longitude and medium-low values for Latitude, which locates the wells in western region of the study area, in a centered position regarding the vertical axis. **Class 2** has the lowest values of Electric Conductivity, median of 1.91 mS/cm, and medium-high values of Static Level with a median of 14.82 m. Longitude values are high, median of -63.90334º, while latitude values are the lowest among all classes, median of 11.11234º, which sets the wells of the class in the southeast. **Class 3** has low values of Electric Conductivity, median of 1.98 mS/cm, and high values of Static Level, median 21.95 m, while the Longitude of the wells is very high, median of −63.899º, and the Latitude medium-low, median of 11.11852º. Thus, the wells in class 3 are in southeast region, above wells in class 2, which shows higher values of latitude. **Class 4** is characterized by high values of Electric Conductivity, median of 3.09 mS/cm, and medium values of Static Level, median of 11.60 m. Meanwhile, Longitude values are high and Latitude values medium, medians of -63.90504º and 11.11995º respectively, which sets the wells in eastern region of the study area, while being centered in vertical axis. **Class 5** has high values of Electric Conductivity, median of 2.94 mS/cm, and medium levels of Static Level, median of 12.23 m, while the Longitude is the highest, median of −63.8938º, the Latitude is the second highest, median of 11.12515º, and so wells from this class are in the northeast corner of the study area. Finally**, Class 6** shows the highest values of Electric Conductivity, with a median of 4.11 mS/cm, and the lowest Static Level, median of 2.04 m. Longitude values are the lowest, despite having a bigger median than Class 1, both minimum and 1st Quartile values are lower. Latitude, on the contrary, follows the opposite trend, with the highest values among all the classes, median of 11.12796º, which locates the wells in the northwest region of the study area. Geolocation of the six classes is shown in Fig. 1.

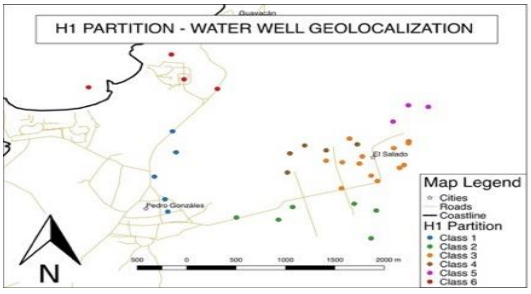


Fig. 1. Geolocation for H1 partition.

Analyzing fig.1, it is possible to see how different classes are grouped in specific regions of the study area. Class 1 is composed by the Water Wells located in Pedro González town urban core, while Class 6 is composed by those Wells surrounding Hesperia Hotel Isla Margarita. The rest of the Classes are located in El Salado Sector. In El Salado Sector Class 2 occupies the southernmost region, Class 5 is located in the northernmost region of the sector, while Class 3 and Class 4 are located between Class 2 and 5, just above El Salado main road.

The results for H2 refer to four profiles/classes discriminated (using Calinski-Harabasz index) as: **Class 1** shows a medium level of Electric Conductivity, median of 2.40 mS/cm, while the Static Level of the wells are medium-low, median of 9.47 m. The wells show low values both for Longitude and Latitude, locating them in the southwest. The wells are in a low Altitude, median of 13 m, with very low Slope, median of 0.7% and medium-low Distance to Sea, median of 1.6 km. Meanwhile, Qualitative Variables show that this class is characterized by having both Camping and Inns, it is in an urban town and the lithology is that of slime. **Class 2** has medium values of Electric Conductivity, median of 2.43 mS/cm, and high values of Static Level, median of 14.82 m. Longitude values are high, while Latitude values are the medium-high, locating the wells in the southeast. The wells in class 2 have high values of Altitude, median of 90 meters, high values of Slope, median of 22.4% and high Distance to Sea, median of 2.64 km. In addition, this class shows no Lodging, Vegetation, mainly thin Brushwood, but also Dense Brushwood and Dense Brushwood in Ridge, and the Lithology is characterized by the existence of all three types of soils, Slime, Granite and Gneiss. **Class 3** has low values of Electric Conductivity, median of 2.04 mS/cm, and very high values of Static Level, median 17.1 m, while the Longitude is very high, and Latitude medium-low, covering the southeast. Besides, the wells have a medium Altitude, 49.5 m, medium Slope, 4.95% and high Distance to Sea, median of 2.56 km. As in class 2, class 3 has no Lodging, the land use is that of Farms in process of Urbanization and the soil is mainly slime, though there are some wells located in granite soil. **Class 4** is characterized by high values of Electric Conductivity, median of 2.77 mS/cm, and very low values of Static Level, median of 2.04 m. Meanwhile, Longitude values are low and Latitude values high, which locates the class in the northeast. In addition, the Altitude of the wells is low, median of 14.5 m, as well as the Slope, median of 0.95%, and low Distance to Sea, median of 0.95 km. The class is strongly characterized by the presence of Resorts, which lead to a land use where different categories, such as Touristic Area, Deforested Desert Surface, and Sea proximity. Finally, the soil is characterized by the presence of Slime. Geolocation of the four classes for H2 partition is shown in Fig. 2.

Observing the geolocation of the wells, discriminated by class, as shown in Fig. 2, it is possible to state that Class 1 gathers the wells located in Pedro Gonzalez town and two wells close to El Salado Sector. Class 2, on the contrary covers the edges in El Salado Sector, which is characterized by low altitude, slopes, and vegetation, which suggest that this class is located in a mountainous area. Class 3 is also related to El Salado Sector. However, its lower altitude and slope, and the presence of many wells located in Farms in Process of Urbanization suggest that class 3 covers the valley bottom of El Salado.

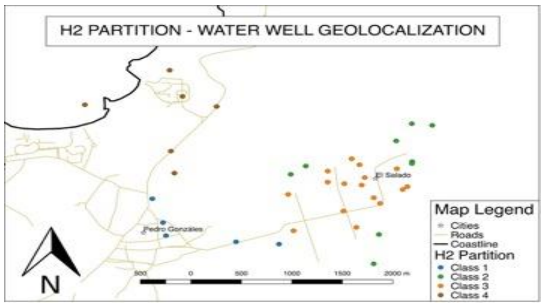


Fig. 2. Geolocation for H2 partition.

Comparing Clustering Results: Since two Clustering Analysis processes have been performed, it is necessary to analyse which one of the two partitions, H1 or H2, explains better the situation. To do so, variables Electric Conductivity and Static Level have been employed, since these two variables show the quality of the water confined in the aquifer. Note that for Electric Conductivity values ranging from 0 mS/cm to 1.2 mS/cm guarantee drinkable water; from 1.2 to 2.5 mS/cm indicate suitable domestic uses (non-drinkable); more than 2.5 mS/cm indicates polluted water or sea water. For Static Level, longer than 20 m are associated with low potential supply wells; from 5 to 20 m indicates risky wells; NE shorter than 5 m corresponds to optimal wells, i.e., appropriated to good supply levels. Ph: commonly drinkable water has values from 6.5 to 8. Analysing the clustering results, H1 partition seems to be a better discriminant both for Electric Conductivity and Static Level. Water quality and level conditions are clearer in H1 partition, as the existence of more classes in H1 allows to have smaller groups which have more homogeneous behaviour within the class. This fact may suggest that the additional information gathered from Open Data sources does not provide any relevant information, as the information of the Experimental Dataset seems to generate a better clustering partition to describe the water quality and the water level of the aquifer. However, this is not the case, as enriching the interpretation of the profiles obtained from partition H1, may provide a better understanding of the aquifer situation and context, regardless of the results in H2 partition.

Final profiling: Illustrative Variables are those variables that enrich the interpretation of the classes of a model, without being part of the construction of the model [11]. Therefore, the variables that are employed as illustrative variables, are the one that were included in the Open Data phase. The illustrative variables included are numerical in the case of Altitude, Slope and Distance to Sea, Average Temperature and are categorical variables for Lodging, Land Use and Vegetation & Lithology. Therefore, the six classes from H1 clustering are enriched in the following way: Class 1 has wells with low Altitude, median of 13 m, low Slope, median of 1.3% and low Distance to Sea, median of 1.12 km. In addition, this class is the only one with inns, as it is in Pedro Gonzalez town, where the soil is slime. Class 2 is characterized by wells with medium Altitude, median of 42 m, a medium Slope, median of 4.65% and a high Distance to Sea, median of 2.59 km. This class is the only one with camping, the Land Use is characterized by the presence of Town Urban Areas and Farming Areas in Urbanization, while the ground is Slime. Class 3 has wells with medium-high Altitude, median of 61 m, medium Slope, median of 7.8%, and high Distance to Sea, median of 2.75 km. The class does not contain any lodging, its land use is Farming Area in Urbanization, while it is the only class with presence of Dense Brushwood and the soil is Slime. Class 4 does have also medium Altitude values, median of 49 m, while the slope is lower than class 3 with medium values, median of 4.3%, and a medium-low Distance to Sea, median of 2.04 km. The class does not have any kind of Lodging, the land use is related also to Farming Area in Urbanization and Thin Brushwood, while the ground has the presence of all types of soil, Slime, Gneiss and Granite. Class 5 has high values of Altitude, median of 96 m, with high Slope values, median of 22.4% and a high Distance to the Sea, median of 2.56 km. The area is characterized by the presence of vegetation, such as, Thin Brushwood and Dense Brushwood in Ridge and the ground consists of Granite. Class 6 has water wells with low values of Altitude, median of 29.5 m, low values of Slope, median of 0.3% and low Distance to Sea, median of 0.4 km. The presence of Resorts characterized the area, as it is a Touristic Area of the seashore.

Open Data used as illustrative variables have enriched the interpretation of the six profiles obtained from H1 partition. This further information allows the expert to better understand external conditions that could be affecting the water quality associated to factors in altitude, slope, distance to the sea, lodging, vegetation, land use and lithology. For instance, proximity to the sea and lithology are key factors to reduce the water quality due to increase in electric conductivity (presence of salty water).

Table II offers a brief of the six profiles in terms of water quality according to the limits recommended by the experts in relation to electric conductivity and static level reported on water wells.

TABLE II. WATER QUALIY FOR FINAL RESULTS

Class	Electric Conductivity	Static Level	Geolocation
1	Medium High: Domestic Use (non-drinkable water)	Low: Optimal supply	West
2	Medium: Domestic Use maybe drinkable water)	Medium: Risky supply	Southeast
3	Medium: Domestic Use (maybe drinkable water)	Medium High: Low potential. Water wells are getting dry	East
4	High: Polluted water or Sea water	Medium: Risky supply	East
5	High: Polluted water or Sea water	Medium: Risky supply	Northeast
6	High: Polluted water or Sea water	Low: Optimal supply	Northwest

Note that most of the water wells does not offer good quality in terms of drinkable-water. In the case of domestic-use only water wells of the class 1 are able to supply the resource due to an efficient static level of the water. Water wells in class 6 has the potential to supply water, but the water quality has been contaminated probably with water from the sea.

If final results (in terms of water quality) are enriched with the illustrative variables, the profile interpretation suggests that:

Class 1 (water wells which can supply water for domestic use) covers Pedro González town's water wells and two wells located in the road that connects the town with Hotel Hesperia Isla Margarita. Pedro Gonzalez town is located in the proximity of seashore in a plain terrain, where slime-based soil is present and town-based lodging activities such as the presence of inns takes place. Attention is required in these wells due to the increase of static level measurements.

Class 2 gathers the wells in the southern area of El Salado valley. These water wells can supply water for drinkable purposes and domestic use, but they have a medium risk to become dry due to the measures of the Static Level. This class is located in a farming area that is undergoing an urbanization process, which can suit the presence of a camping. The use of the soil can explain the decreasing potential to supply water.

Class 3 comprises wells that are located in the northeast area of El Salado valley, in a higher altitude than the wells in class 2. These wells can supply water, but they are reaching the limit to become dry. The Static Level is high, thanks probably to the altitude mentioned before. The area in which this class is located covers a transition area between virgin mountains and farming urbanizing area, reason why the land use of the class consists of farming areas in urbanization in combination with dense brushwood, being the only class in which this last category of vegetation is present.

Classes 4, 5 and 6 are composed of water wells with no capability to supply drinkable-water or water for domestic use. Their levels of electric conductivity imply that water is polluted or contaminated with salt. Water supply at these places can be only used for non-human activities.

Class 4 as class 3 is located in a transition between El Salado Valley and the mountain range that surrounds the valley, with a medium-high altitude and medium slope. Electric Conductivity is high, while the Static Level shows medium values. This class has no lodging, the land is used for farming in urbanizing area, and there is also vegetation thanks to the presence of thin brushwood. In addition, the semi mountainous nature of the terrain implies the presence of both slime and rocky soil in which Gneiss and Granite are present.

Class 5 is located in a mountainous area in the northeastern part of El Salado sector, with high altitude, slope, and distance to the sea. The Electric Conductivity is high, while the Static Level remains in the medium values. Besides, only vegetation is present in this class, thin brushwood and dense brushwood in ridge and the ground consists of granite.

Class 6 is located in the northwestern corner of the study area, in the surroundings of Hotel Hesperia Isla Margarita. This hotel is a resort, located by the seashore and therefore altitude, slope and distance to sea are low. This low altitude also generates a low Static Level, whereas the Electric Conductivity holds the highest values between all the classes. Land use in this area is related to tourism and the lithology is characterized by the presence of the sea. The extensive use of water by the Hotel and the proximity of the sea have caused that water quality around this zone has got worse.

Once these profiles were analyzed, it is possible to observe that there are only two groups (class 2 and 3 around the Salado Valley) with the capability of supply water for human and domestic use, but they are in a risky situation because of their static level values. Class 1 has still potential to supply water, but only for domestic activities and not for drinking use. Class 6 is a consequence of the extensive use of the water due to tourism and big hotels (zones close to Pedro Gonzalez town and the coast), where water wells do have water but contaminated by the sea or the tourism activities. In fact, water wells in class 6 are also affected by the existence of saline intrusion due to their proximity to the seashore. Finally, a strange phenomenon happens in classes 4 and 5, as both of them are classes with a significant presence of vegetation and less human influence, thanks to their location in a mountainous area. Deeper analysis should be conducted at this point in order to clarify why these wells present a scarce water quality.

V. CONCLUSIONS AND FUTURE WORKS

This paper has addressed the analysis of Pedro Gonzalez-El Salado Aquifer in Isla Margarita, Venezuela, employing Data Mining techniques over two databases. On the one hand an experimental database (Electric Conductivity, Static Level, and pH) consisting of information related to water quality, the geolocation and the year of the measurement has been used, while in a second step different variables from Open Data sources have been added (Altitude, Slope, Distance to Sea, Lodging, Land Use and Vegetation and Lithology). After analyzing both databases a Hierarchical Clustering process of classification has been conducted, obtaining a first partition of six classes(H1) and a second partition of four(H2), respectively. Analyzing the differences between both partitions, H1 partition has shown to be more discriminant for Electric Conductivity and Static Level.

Nevertheless, H1 though being a better descriptor of the situation of the aquifer, it did not explain any reasons for the differences, reason why it was decided to employ the variables obtained from open data source as context illustrative variables for explaining partition H1and to improve the characterization of the profiles obtained from H1.

This enrichment of interpretation showed that class 2 and 3 could have lower Electric Conductivity thanks to the direction of the recharge and the confinement of the aquifer which can limit impacts related to the farming that takes place in the area. Moreover, Electric Conductivity value of these classes have lower values than the rest of the classes for being also in the direction of recharge and receiving more fresh water. On the contrary, class 6 shows the highest values of Electric Conductivity thanks to saline intrusion in the aquifer which can be generated by the excessive extraction of water use from the Hotel Hesperia Isla Margarita and its proximity to the sea. Finally, class 4 and 5 show also high values of Electrical Conductivity which can be caused by the higher superficiality of the water present in the wells of these classes. However, since these two classes are characterized

for its vegetation and inferior human impact, further research should be conducted to stablish what is causing this higher value of Electric Conductivity. By this way, this paper shows that it is convenient to use experimental variables as interest variables (water quality), while context variables are employed to illustrate and enrich both the analysis and the interpretation of the profiles.

Data Science allowed to characterize the situation of the aquifer's water wells. This characterization helps to support strategies, decisions and interventions associated to water management. This is since it is possible to formulate hypothesis that explain the differences between Electric Conductivity and Static Level between classes and understand in which direction act the differences for each profile.

Finally, it is important to state that to improve future works of this research, it is convenient to review the design of Lodging variable, which should better stablish its values not by proximity, but for the location of the well up or downstream of the lodge. In addition, it would also be convenient to have better climatic data, given the difficulties that have risen for finding information of precipitation, relative humidity, or different measurements of temperature in Pedro Gonzalez area.

REFERENCES

[1] United Nations, Goal 6: Ensure access to water and sanitation for all, obtained from Sustainable Development Goals, http://www.un.org/sustainabledevelopment/water-and-sanitation/, September 2015.

[2] C. Fry, The Impact of Climate Change: The World's Greatest Challenge in the Twenty-first Century. New Holland Pub Limited, 2008.

[3] A. Sahuquillo, "La importancia de las aguas subterraneas", Rev.R. Acad.Cienc.Exact.Fís.Nat, vol. 103(1), pp. 97-114, 2009.

[4] Asamblea Nacional de Venezuela, Primer Informe: Problemática de Agua en Venezuela, Asamblea Nacional de la República Bolivariana de Venezuela - Comisión Mixta Especial para Investigar la Problemática del Agua. Caracas, 2016.

[5] B. Akin, T. Atici, H. Katircioglu, and F. Keskin, "Investigation of water quality on Gokcekaya dam lake using multivariate statistical analysis," Environ Earth Sci, vol. 63, pp. 1251-1261, 2011.

[6] I.Yeung," Multivariate analysis of the Hong Kong Victoria harbour water quality data," Environmental Monitoring and Assessment, vol. 59, pp. 331-342, 1999.

[7] K. Gibert, D. Conti, and D. Vrecko, "Assisting the end-user in the interpretation of profiles for decision support: an application to wastewater treatment plants," Environmental Engineering and Management Journal , vol. 11 (5), pp. 931-944, 2012.

[8] K. Gibert, G. Rodriguez Silva, and I. Rodriguez Roda, "Trajectories Mining Between Subprocesses in a Wastewater Treatment Plant". 4th International Congress on Environmental Modelling and Software – Barcelona, Spain, July 2008 .

[9] A. Candelieri, D. Conti, and F. Archetti, "Improving Analytics in Urban Water Management: A Spectral Clustering-based Approach for Leakage Localization," Procedia - Social and Behavioral Sciences, vol. 108, pp. 235-248, 2014.

[10] T. Guo, and J. Englehardt, "Principles for scaling of distributed direct potable water reuse systems: A modeling study," Water Research, vo. 75, pp. 146-163, 2015.

[11] L. Lebart, A. Morineau, and J. Fenelon, Traitement statistique des données. Paris Dunod, 1990.

Análisis descriptivo de cáncer de mama usando minería de datos

Antonio Manuel Trasierras
Dept. de Informática y Análisis Numérico,
Universidad de Córdoba,
Córdoba, España
sc2trfra@uco.es

José María Luna y Sebastián Ventura
Dept. de Informática y Análisis Numérico,
Instituto Interuniversitario de Investigación en
Data Science and Computational Intelligence (DaCSI)
Universidad de Córdoba, Córdoba 14071
Email: {jmluna, sventura}@uco.es

***Resumen*—Este trabajo presenta una aproximación basada en *emerging pattern mining* para el análisis genómico del cáncer. A diferencia de la mayoría de las propuestas actuales basadas en aproximaciones predictivas, nuestra propuesta trata de mejorar el conocimiento de la enfermedad de manera descriptiva, sin considerar hipótesis de partida ni conocimiento previo. La metodología propuesta es capaz de obtener relaciones de alto orden para poder explorar relaciones génicas directas e indirectas que pueden influenciar diferentes rutas funcionales relacionadas con la enfermedad. La idea principal de la propuesta es que la división del conjunto de datos genómicos en dos subconjutos pertenecientes a sanos y enfermos, nos permitirá obtener el conjunto de genes y relaciones intergénicas responsables de la enfermedad. La utilidad de la propuesta se pone a prueba evaluando un conjunto de muestras pareadas de cáncer de mama en formato *RNA-Seq*. Algunos de los resultados han sido validados en la literatura mientras que otros podrían describir nuevas relaciones funcionales entre diferentes genes.**

***Index Terms*—rna-seq, cancer, emerging-patterns, data-mining, bioinformatics**

I. Introducción

En la actualidad existen multitud de investigaciones relacionadas con el cáncer, principalmente a causa de la gran incidencia que presenta en todo el mundo [1]. Sin embargo, tanto el análisis del cáncer como la personalización de terapias no son tareas triviales. Desde un punto de vista analítico, los estudios sobre la enfermedad suelen estar caracterizados por un gran número de genes, del orden de decenas de miles frente a pocas centenas de muestras en el mejor de los casos. Este problema, conocido comúnmente como *fat-short problem* [2] y el desbalanceo de clases propio de datos biológicos añade complejidad al correcto análisis estadístico de las muestras.

Hoy en día, la mayoría de los estudios moleculares para la detección de biomarcadores en datos transcriptómicos de cáncer siguen un flujo de trabajo estandarizado [3]. En primer lugar se lleva a cabo un análisis exploratorio de los datos para comprobar su distribución y detectar posibles outliers. Para esta tarea se suelen considerar técnicas clásicas como el análisis de componentes principales y clúster. Acto seguido, para determinar el conjunto de genes implicados en la enfermedad, se llevan a cabo análisis de expresión diferencial para detectar genes cuya expresión varía entre las condiciones del estudio (por ejemplo, sano frente a cáncer). Posteriormente, para estudiar las relaciones presentes entre estos genes y una condición dada, se realizan análisis de correlación con los que posteriormente pueden construirse redes de coexpresión génica. Finalmente, con el objetivo de mejorar la interpretabilidad final sobre el conjunto de genes obtenido, se llevan a cabo un análisis de enriquecimiento funcional utilizando la información contenida en grandes bases de datos biológicas. Sin embargo, aunque comúnmente usado, este flujo de trabajo implica una clara desventaja: las interacciones entre los distintos genes presentes en el estudio sólo suelen ser consideradas mediante pares, mientras que las relaciones de alto orden permanecen poco estudiadas. No obstante, hay cada vez más evidencia de que los genes interactúan entre sí en órdenes superiores a relaciones por pares [4].

La creciente popularidad de la minería de datos ha propiciado el desarrollo de varios estudios para la detección temprana del cáncer basados en *machine learning* (ML) [5]. Por ejemplo, recientemente se ha abordado el diagnóstico temprano de melanoma mediante técnicas de Deep Learning [6], así como considerando factores de riesgo [7]. Sin embargo, la mayoría de propuestas actuales se centran en tareas predictivas para la detección temprana del cáncer. El presente estudio, en cambio, se basa en el uso de técnicas descriptivas de minería de patrones, conocidas por ser capaces de proveer al investigador de un conocimiento altamente interpretable. Entender qué conjuntos de genes y las relaciones subyacentes que se dan entre ellos es clave para la búsqueda de biomarcadores contra la enfermedad. Con este fin, este trabajo presenta una metodología basada en el uso de una técnica de minería de patrones descriptiva: *emerging patterns* (EP). Esta tarea trata de identificar el conjunto de patrones discriminativos entre dos clases presentes en los datos. Mediante la presente metodología, los datos son divididos en dos subconjuntos, uno para cada condición (sano y enfermo), un diseño experimental típico en bioinformática. De este modo, se extrae el conjunto de patrones cuya frecuencia varía de manera significativa entre los subconjuntos anteriormente mencionados. El principal objetivo que se persigue en este trabajo es demostrar que, mediante el estudio de datos transcriptómicos de cáncer, la metodología planteada es capaz de extraer conocimiento útil en la forma de relaciones génicas de alto orden intrínsecas a cada condición estudiada.

Las principales contribuciones de este trabajo pueden resu-

mirse como:

- Propuesta de una metodología capaz de considerar relaciones de alto orden. Esta metodología permite la extracción de variables pertenecientes a diferentes rutas de señalización así como determinar relaciones entre el conjunto de genes vinculados a la enfermedad.
- La propuesta permite la extracción de conocimiento útil sin considerar ningún tipo de conomiento ni hipótesis previas.
- Una técnica posterior de filtrado que evita la redundancia en los resultados típicamente producida por técnicas descriptivas, mejorando la interpretabilidad y accionabilidad de los resultados obtenidos.

La metodología propuesta se ha aplicado sobre datos de cáncer de mama extraídos de *The Cancer Genome Atlas* (TCGA) [8]. Los resultados obtenidos demuestran la utilidad de la propuesta para la búsqueda de biomarcadores moleculares en cáncer. Algunos de los resultados obtenidos han sido descritos en la literatura como biomarcadores tumorales mientras que otros no han sido descritos aún, pudiendo suponer un punto de partida para futuros estudios experimentales.

El artículo se divide en diferentes secciones: la Sección II presenta unas nociones básicas sobre la tarea de EP y algunos estudios sobre cáncer. La Sección III describe la metodología propuesta mientras que en la Sección IV se presenta un análisis exhaustivo sobre datos de secuenciación masiva de cáncer de mama. Finalmente, la Sección V remarca unas breves conclusiones.

II. Preliminares y literatura previa

Esta sección define formalmente la tarea de EP y repasa algunas de las aplicaciones previas de ML y minería de datos al estudio del cáncer.

II-A. Emerging pattern mining

En minería de patrones [9], la unidad fundamental de conocimiento es el patrón. Un patrón P es un conjunto de items relacionados entre sí, pertenecientes a una base de datos Ω. Formalmente, un patrón P puede ser definido como $P \subseteq I \in \Omega$ donde I es el conjunto de n items $\{i_1, i_2, ..., i_n\}$ pertenecientes a la base de datos Ω. La tarea de minería de patrones trata de encontrar el conjunto de patrones presente en al menos una fracción de los registros de una base de datos. La métrica de calidad comúnmente utilizada en esta tarea es el soporte o frecuencia. En términos absolutos, el soporte de un patrón P es el número de registros donde P está incluido. El soporte también puede definirse en términos relativos, considerando el porcentaje de registros que contienen a P. Los registros se conocen comúnmente como transacciones, de este modo, se define $T = \{t_1, t_2, ..., t_m\}$ como el conjunto de todas las transacciones presentes en Ω. Formalmente, el soporte de un patron P en una base de datos Ω se define como:

$$Support\ (P, \Omega) = \frac{|\forall t \in T : P \subseteq t, t \subseteq I|}{|T|} \quad (1)$$

Emerging pattern mining (EPM) es una técnica perteneciente al grupo de técnicas de minería de patrones descriptivas [9], la cual se caracteriza por buscar patrones discriminativos (aquellos patrones cuyo soporte varía considerablemente entre dos grupos). Dicho de otro modo, esta técnica trata de buscar patrones que distingan claramente los grupos considerados. Consideremos I como el conjunto de items pertenecientes a dos grupos de datos Ω_1 y Ω_2. Consideremos, también, que estos grupos de datos tienen un formato tabular-binario (es decir, un item puede o no estar en los datos) y que $T = \{t_1^1, t_2^1, ..., t_n^1\}$ es el conjunto de transacciones presentes en los registros del grupo Ω_1 tal que $\forall t^1 \in T : t^1 \subseteq I \in \Omega_1$. Se considera un patrón P como *emerging pattern* si su soporte varía considerablemente entre el grupo Ω_1 y Ω_2. La métrica comúnmente considerada para medir el interés de este tipo de patrones es el *Growth Rate* (GR) que se define como el ratio entre el soporte de un patrón P entre el grupo Ω_1 y su soporte en el grupo Ω_2 (ver Ecuación 2).

$$GR\ (P) = \frac{Support\ (P, \Omega_1)}{Support\ (P, \Omega_2)} \quad (2)$$

Dependiendo del valor de GR, los patrones pueden clasificarse en diferentes categorías [9] como *minimal, maximal, essential,* etc. Dentro de estas categorías, una de las más comúnmente usadas son los *jumping emerging patterns* (JEPs). Un patrón P es un JEP si aparece al menos una vez en el grupo Ω_1 y ninguna en el grupo Ω_2. De esta manera, los JEPs presentan un GR infinito y son capaces de describir características exclusivas pertenecientes a cada grupo considerado.

II-B. Data science en bioinformática

En los últimos años se han desarrollado numerosas propuestas de ML para el estudio del cáncer gracias, en parte, al desarrollo de nuevas tecnologías de secuenciación y al auge de la minería de datos. Una gran parte de estas aplicaciones están centradas en predecir o detectar el cáncer en sus etapas de desarrollo más tempranas, como el uso de Deep Learning para la detección de melanoma a través del análisis de imágenes [6]. Por otra parte, cada vez son más los estudios predictivos sobre distintos tipos de cáncer que prestan especial atención a cómo se presenta el conocimiento al usuario final. En este sentido, varias técnicas descriptivas han sido combinadas para producir clasificadores entendibles y precisos bajo el marco de la tarea conocida como clasificación asociativa [10]. Por ejemplo, Vengateshkumar [11], construyó un clasificador usando reglas de asociación booleanas para el estudio del cáncer usando como método de selección de variables el test estadístico de la T de Student. El uso de EP también ha sido considerado para generar clasificadores capaces de predecir distintos tipos de cáncer, como leucemia linfobástica [12] y cáncer de colon sobre datos de *microarray*. Los datos provenientes de *microarrays* fueron discretizados mediante un método basado en la entropía de manera que los valores de expresión génica fueron divididos en intervalos de valores disyuntivos para los que el valor de la entropía es mínimo. De manera similar, los autores de este último

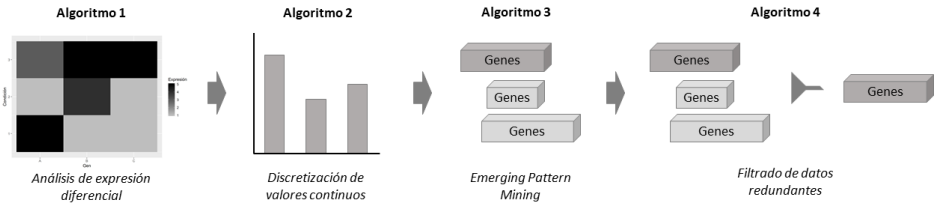


Figura 1: Flujo de trabajo de la metodología propuesta

trabajo utilizaron los cambios a nivel de expresión génica para identificar genes con buena prognosis en leucemia y cáncer de colón a través del uso de clasificadores [13].

Aún cuando la mayoría de propuestas de minería de datos han centrado su investigación en el desarrollo de modelos predictivos, las aportaciones basadas en metodologías descriptivas se están volviendo cada vez más importantes. Recientemente, algunos autores [14] propusieron una aproximación descriptiva basada en la extracción de reglas de asociación booleanas para describir relaciones génicas presentes en datos de *microarrays*. Los autores usaron metodologías de inferencia estadística clásicas basadas en el uso de T-tests en lugar de utilizar técnicas de referencia en la comunidad bioinformática como el uso de modelos lineares [15], pudiendo introducir sesgos en el análisis posterior.

III. Una metodología de análisis basada en Emerging Pattern Mining

El uso de una metodología descriptiva basada en EPM tiene un gran interés para el estudio del cáncer ya que permite la descripción de características y relaciones de alto orden que diferencian claramente dos grupos sin requerir de ningún tipo de conocimiento o hipótesis de partida. A diferencia de la mayoria de aproximaciones actuales, basadas en el uso de clasificadores, esta propuesta trata de generar información descriptiva útil e interpretable, la cual es de vital importancia para el desarrollo de nuevas terapias personalizadas. De este modo, esta sección trata de introducir la metodología propuesta para el análisis de datos de cáncer. La Figura 1 ilustra de manera esquemática el flujo de trabajo de dicha metodología, la cual consta de cuatro procedimientos.

El primer procedimiento de la metodología propuesta (ver Algoritmo 1) es el encargado de seleccionar las variables génicas que presentan una expresión diferencial significativa entre las dos condiciones de estudio (es decir, cáncer y sano). Como es común en la mayoría de estudios genómicos, con objeto de minimizar el error en la estimación de los genes diferencialmente expresados es necesario eliminar aquellos que no contribuyen al análisis del cáncer al no presentar expresión en el estudio. No hay interés en analizar genes que no se expresan en al menos una cierta cantidad de muestras pertenecientes a cualquiera de los grupos considerados. Por lo tanto, el primer paso para determinar los genes a analizar es establecer un punto de corte basado en los conteos por cada millón de *reads* mapeadas (CPM) de cada gen (ver Algoritmo 1, líneas 3 a 5). Una *read* es una secuencia bruta

Algorithm 1 Pseudocódigo para la selección de genes

Entrada Datos de *RNA-Seq* incluyendo dos tipos de muestras

Salida Genes diferencialmente expresados (Conjunto deg)

1: **procedure** Análisis de expresión diferencial
2: **for** cada gen g en los datos **do**
3: **if** $CPM < 1$ a lo largo de los grupos **then**
4: Eliminar g de los datos
5: **end if**
6: **end for**
7: $nGenes \leftarrow$ calcular el número de genes restantes en los datos
8: **if** $nGenes > 0$ **then**
9: $normData \leftarrow$ Realizar una normalización TMM sobre los datos usando edgeR
10: $deg \leftarrow$ Obtener el conjunto de genes diferencialmente expresados de $normData$ mediante los tests considerados en edgeR con un $FDR <= 0{,}05$
11: **end if**
12: **return** deg
13: **end procedure**

de ARN producida por las máquinas de secuenciación, la cual puede constar de múltiples segmentos. En este caso, asumimos que un gen es relevante si tiene al menos 1 CPM a lo largo de las muestras de cualquier condición del estudio. Dicho de otro modo, asumimos que un gen debe tener al menos un CPM ≥ 1 en al menos N muestras, siendo N el número de muestras del grupo más pequeño considerado. Entonces, los conteos de los genes determinados como relevantes son normalizados mediante la media recortada de los valores M (TMM) para tratar la variación en el número total de conteos de genes entre las muestras de interés. Tras esto, se lleva a cabo un análisis de expresión diferencial mediante el paquete de análisis estadístico edgeR [16] (ver Algoritmo 1, línea 10) considerando una tasa de falsos descubrimientos (FDR) de 0.05. Por último, el conjunto de los genes diferencialmente expresados entre ambas condiciones así como sus conteos normalizados son devueltos y usados como entrada para el siguiente paso de la metodología.

Como nota adicional, este método de selección de características no presenta los sesgos comúnmente considerados por otros procedimientos [13], [14]. Por ejemplo, a diferencia de los métodos de selección de características clásicos como

Algorithm 2 Pseudocódigo para la discretización

Entrada Conjunto de genes diferencialmente expresados (Conjunto deg)
Salida Datos discretizados (Conjunto deg')

1: **procedure** DISCRETIZACIÓN
2: $deg' \leftarrow$ Matriz transpuesta de los datos de deg
3: **for** cada gen g en el conjunto deg' **do**
4: $g \leftarrow$ Discretiza los valores de expresión de g en 5 etiquetas
5: **end for**
6: **return** deg'
7: **end procedure**

la entropía o la selección de características basada en la correlación, el procedimiento presentado considera cualquiera de los genes participantes en rutas de señalización secundarias, sin pasar por alto genes secundarios que contribuyen al estado total de activación de rutas de señalización principales y/o secundarias. Por otro lado, la metodología propuesta no considera el uso de simples tests estadísticos de la T clásicos, los cuales pueden introducir sesgos en el análisis debido a la asumpción de la distribución normal. En cambio, el método propuesto sigue una distribución binomial negativa dado que los datos de *RNA-Seq* están principalmente basados en conteos (no existen valores negativos ni decimales).

El segundo procedimiento de la metodología (ver Algoritmo 2) comprende la discretización de los valores de expresión de los genes. En primera instancia, el *dataset* con los valores de expresión normalizados de los genes diferencialmente expresados obtenido en el paso anterior se transpone haciendo que los genes tomen las columnas y las muestras las filas (ver Algoritmo 2, línea 2). Entonces, este procedimiento discretiza los niveles de expresión de conteos en 5 etiquetas diferentes, las cuales corresponden a la división equitativa del rango intercuartílico de sus niveles de expresión: EA (extremadamente alto), A (alto), M (medio), B (bajo) y EB (extremadamente bajo). Este es un paso opcional ya que los datos pueden estar discretizados previamente (o usarse una aproximación diferente). Finalmente, se devuelve el conjunto de genes discretizados *deg'* para ser usado como entrada del siguiente procedimiento (ver Algoritmo 2. Línea 6).

El tercer procedimiento corresponde a la parte de EPM (ver Algoritmo 3). Como entrada, este procedimiento toma el conjunto de los valores de conteo de los genes previamente discretizados y divide el *dataset* en dos subconjuntos: muestras cancerosas y muestras sanas. De este modo, para cada uno de los subconjuntos, el procedimiento extrae todos sus patrones frecuentes considerando un punto de soporte mínimo α, predefinido por el usuario (ver Algoritmo 3, línea 4). Cuando todos los patrones han sido obtenidos, el procedimiento sólo conserva los JEPs de cada uno de los grupos (ver Algoritmo 3, líneas 5 a 11). De este modo el algoritmo es capaz de analizar patrones intrínsecos de cada una de las clases consideradas, tanto propias del cáncer como de los tejidos sanos.

Algorithm 3 Pseudocódigo para la extracción de EPs

entrada Conjunto de genes discretizados (Conjunto deg')
Salida Todos los JEPs para cada grupo G (JEP_G)

1: **procedure** MINERÍA DE PATRONES FRECUENTES (ALGORITMO LCM)
2: **for** cada grupo G en deg' set **do**
3: $X_G \leftarrow$ muestras de deg' pertenecientes al grupo G ▷ Crear un *subdataset* X_G para cada grupo G
4: $P \leftarrow \emptyset$ ▷ El conjunto de los patrones frecuentes de G
5: $JEP_G \leftarrow \emptyset$ ▷ El conjunto de los JEPs de G
6: $P \leftarrow$ Aplicar algoritmo LCM en X_G con un soporte mínimo α
7: **for** cada patrón frecuente p en P **do**
8: **if** p is JEP **then**
9: $JEP_G \leftarrow JEP_G \cup p$
10: **end if**
11: **end for**
12: **end for**
13: **return** JEP_G, X_G
14: **end procedure**

La parte correspondiente a la extracción de patrones frecuentes incluida en este procedimiento (ver Algoritmo 3, línea 6) se lleva a cabo mediante el algoritmo LCM (*linear closed itemset mining*) [17], uno de los algoritmos más rápidos y eficientes propuestos hasta la fecha.

El cuarto y último procedimiento (ver Algoritmo 4) trata de reducir el número de soluciones redundantes producidas comúnmente en tareas descriptivas. En este sentido, dada la gran cantidad de genes considerados, el procedimiento busca mantener un conjunto acotado de soluciones representativas. Este conjunto debe incluir características capaces de describir a las muestras de cada uno de los subconjuntos considerados (preferentemente a todas ellas) con un número de soluciones pequeño que facilite la interpretabilidad de los resultados. Para ello, el procedimiento primero ordena las soluciones de acuerdo a su soporte en orden descendente (ver Algoritmo 4, línea 3). Si se produce algún empate, se posiciona primero la solución que contenga un mayor número de genes como se muestra de la línea 4 a la 8 en el Algoritmo 4. Entonces, se obtiene la primera solución de este ranking y se marcan todos los registros cubiertos por ella (ver Algoritmo 4, líneas 10 a 12). Este procedimiento se repite solución por solución hasta que todos los registros han sido cubiertos o no queda ninguna solución por seleccionar (ver Algoritmo 4, líneas 13 a 18). Este último procedimiento considera un valor de umbral de solapamiento β definido por el usuario. Este umbral se asegura de al menos un β % de nuevos registros no marcados deban ser cubiertos por las soluciones. Adicionalmente, se considera un parámetro θ para asegurar un número mínimo de genes en los patrones obtenidos. Por defecto este parámetro tiene un valor de 2.

Algorithm 4 Pseudocódigo para filtrar soluciones redundantes

Entrada JEP_G, X_G, β, θ
Salida Res ▷ Conjunto representativo de JEPs de un grupo G.

1: **procedure** FILTRAR SOLUCIONES REDUNDANTES
2: $JPE_G \leftarrow$ Elimina los patrones de JPE_G cuyo número de genes sea menor a θ
3: $JPE_G \leftarrow$ Ordena JPE_G en orden descendente de soporte
4: **for** cada conjunto de soluciones $S \subseteq JPE_G$ que tengan el mismo soporte **do**
5: $JPE_G \leftarrow JPE_G \setminus S$
6: $S \leftarrow$ Ordena S en orden descendente de longitud
7: $JPE_G \leftarrow JPE_G \cup S$ ▷ Añade S a la misma posición que tenía originalmente (Ranking de soporte)
8: **end for**
9: $Res \leftarrow \emptyset$
10: $p \leftarrow$ Toma el primer elemento de JEP_G
11: $X_G \leftarrow$ marca los registros cubiertos por p
12: $Res \leftarrow Res \cup p$
13: **for** cada p en JEP_G **do**
14: **if** p cubre al menos $\beta\,\%$ nuevos registros no marcados **then**
15: $Res \leftarrow Res \cup p$
16: $X_G \leftarrow$ marca los registros cubiertos por p
17: **end if**
18: **end for**
19: **return** Res
20: **end procedure**

IV. Caso de estudio sobre datos de cáncer de mama

Esta sección busca ilustrar el uso y utilidad de la metodología propuesta a través del análisis de un caso de estudio con muestras de cáncer de mama obtenidas de la base de datos *The Cancer Genome Atlas* (TCGA) [1]. El cáncer de mama es el cáncer que más incidencia presenta sobre las mujeres, presentando un impacto terrible en la sociedad. A través del uso de la metodología propuesta esperamos encontrar conocimiento ya reflejado en la literatura además de relaciones nuevas o poco estudiadas.

El formato de los datos son conteos estimados mediante RSEM, un tipo de dato que puede ser usado como entrada por paquetes de análisis estadístico para *RNA-Seq* como edgeR. Únicamente se tomaron muestras pareadas (cancer vs tejido adyacente sano). Tras la aplicación de la metodología propuesta se llevó a cabo un análisis de enriquecimiento funcional y una revisión de la literatura sobre las soluciones obtenidas para mejorar la comprensión biológica de los datos.

El *dataset* analizado comprende 224 muestras pareadas con 20,531 genes. El primer procedimiento de la metodología redujo el número de genes con una expresión diferencial significativa a 11,738. Después, este conjunto de genes fue discretizado en 5 intervalos (consultar segundo procedimiento de la metodología propuesta) y el algoritmo basado en EPM fue aplicado sobre los dos conjuntos de muestras. Los umbrales de soporte mínimo considerados para cada *dataset* fueron de 30 % y 35 % para cada uno de los grupos considerados, obteniendo un total de 167,239 y 468,708 patrones frecuentes para el grupo de cáncer y el grupo sano, respectivamente. Posteriormente, el procedimiento de filtrado se aplicó sobre este conjunto de soluciones dando lugar a un conjunto de 15 soluciones para cada grupo de manera respectiva [2]. Ambos grupos de soluciones representan el 87.5 % y 83.93 % del conjunto de muestras de cáncer y tejido adyacente sano, respectivamente. Algunos de los patrones obtenidos se describen a continuación gracias a la información proporcionada por el análisis de enriquecimiento funcional y la revisión bibliográfica.

Analizando el primero de los patrones obtenidos {CCNB1 = EA, CDK1 = EA, KIF23 = EA, NUSAP1 = EA, CCNB2 = EA, MELK = EA, NCAPG = EA, BUB1B = EA, NCAPH = EA, C12orf48 = EA} (ver patrón con ID 1, en el anexo web) el análisis de enriquecimiento funcional reveló que todos los genes excepto C12orf48 juegan un papel importante en el ciclo celular actuando como oncogenes en cáncer de mama o en otros tipos de cáncer [18]. C12orf48 codifica una proteína de unión a PARP-1 llamada PARPBP cuya función es la inhibición del mecanismo de reparación del ADN a través de la recombinación homóloga. Varios estudios han sugerido la up-regulación de este gen en varios tipos de cáncer como el cáncer hepatocelular [19], sugiriendo además su posible rol como oncogén dentro del ciclo celular. De este modo, es posible que este gen comparta un marco de regulación común con el resto de genes del patrón analizado. Este es un claro ejemplo del potencial del uso de la minería de patrones bajo esta metodología para llegar a conocimiento útil sin el requerimiento de conocimento previo o hipótesis de partida.

Por otro lado, el patrón con ID 8 (ver primera tabla en anexo web) {LDB2 = EB, CXorf36 = EB, MMRN2 = EB, ECSCR = EB, MYCT1 = EB, CLEC14A = EB} muestra otro resultado interesante obtenido mediante la metodología propuesta. Los genes incluidos en este patrón han sido detectados en un estudio reciente [20], denotando una relación con funciones endoteliales. Es interesante recalcar que la función de CXorf36 permanece poco estudiada mientras que la de CLEC14A parece ser la up-regulación de actividades angiénicas y de migración celular [21]. La relación existente entre los genes implicados en este patrón podría suponer un punto de partida para el desarrollo de posibles terapias antiangiogénicas en cáncer de mama.

Finalmente, es importante señalar algunos de los patrones pertenecientes a la clase de tejido adyacente sano. En este sentido, el análisis del patrón (ID 17, anexo web) {HSPB6 = EA, KCNIP2 = EA, AOC3 = EA, LIPE = EA, GYG2 = EA, AQP7 = EA, GPD1 = EA, TMEM37 = EA, HEPN1 =

[1]Los datos pueden ser descargados en https://www.cancer.gov/about-nci/organization/ccg/research/structural-genomics/tcga

[2]El conjunto de soluciones obtenidas puede ser consultado en la página web http://www.uco.es/kdis/analisis-descriptivo-de-cancer-de-mama-usando-mineria-de-datos/.

EA, AIFM2 = EA, CIDEC = EA, PNPLA2 = EA, TYRO3 = EA, SGCG = EA, ALDH2 = EA} revela que varios de sus genes están implicados en la desregulación del metabolismo de lípidos en cáncer (LIPE, GYG2, AQP7, GPD1, CIDEC y PNPLA2). La falta de lipasas en cáncer de mama se asocia con métodos alternativos de obtención de energía por parte de las células tumorales como el metabolismo de la glucosa y de la glutamina. Estos resultados concuerdan con otros hallazgos ya discutidos en la literatura [22] donde se detalla que la desregulación del metabolismo lipídico a través de la ausencia de lipasas y ácidos grasos libres así como una baja actividad de la señalización PPAR-γ son eventos clave en la formación de varios tipos de cáncer como los liposarcomas. Es interesante resaltar que, mientras que algunos de estos genes como HSPB6 han sido descritos por separado como supresores tumorales en cáncer de mama [23], no se ha descrito ningún nexo de interacción entre los genes relacionados con el metabolismo lipídico y el resto de genes del patrón.

V. Conclusiones

En el presente trabajo se propone una metodología de análisis del cáncer basada en el uso de EPM. Esta metodología consta de cuatro procedimientos especialmente diseñados para el análisis de datos *RNA-Seq*, los cuales permiten la extracción de conocimiento altamente interpretable a partir de datos de cáncer. La metodología permite trabajar sin hipótesis ni conocimiento previo, considerando además relaciones de alto orden entre los genes responsables de la enfermedad. En este sentido, esta propuesta presenta una mayor robustez que otras aproximaciones que sólo consideran relaciones a pares o asumen homocedasticidad, permitiendo en última instancia un análisis descriptivo más potente. Para corroborar la metodología, se ha desarrollado un caso de estudio práctico sobre datos de cáncer de mama extraídos de *The Cancer Genome Atlas* (TCGA), obteniendo conocimiento previamente contemplado en la literatura que permite validar la adecuación de la misma. Además, se han detectado posibles nuevas relaciones entre genes responsables del metabolismo lipídico en este cáncer. Por estos motivos creemos que el uso de EPM y en especial de esta metodología debería ser considerado en la búsqueda de biomarcadores moleculares en cáncer.

Finalmente, nótese que esta metodología es extensible al análisis de cualquier enfermedad siempre que se presente un análisis de tipo binario (sanos frente a enfermos), pudiendo ser aplicada de este modo al estudio de enfermedades coronarias o neurodegenerativas, entre otras, que afecten al transcriptoma de los pacientes.

Acknowledgment

Este trabajo fue financiado por la Universidad de Córdoba, proyecto UCO-FEDER 18 REF.1263116 MOD.A. También ha sido financiado por el Fondo Europeo de Desarrollo Regional.

Referencias

[1] H. Sung, J. Ferlay, R. L. Siegel, M. Laversanne, I. Soerjomataram, A. Jemal, and F. Bray, "Global cancer statistics 2020: Globocan estimates of incidence and mortality worldwide for 36 cancers in 185 countries," *CA: a cancer journal for clinicians*, vol. 71, no. 3, pp. 209–249, 2021.

[2] S. L. Pomeroy, P. Tamayo, M. Gaasenbeek, L. M. Sturla, M. Angelo, M. E. McLaughlin, J. Y. Kim, L. C. Goumnerova, P. M. Black, C. Lau *et al.*, "Prediction of central nervous system embryonal tumour outcome based on gene expression," *Nature*, vol. 415, no. 6870, p. 436, 2002.

[3] M. I. Love, S. Anders, V. Kim, and W. Huber, "Rna-seq workflow: gene-level exploratory analysis and differential expression," *F1000Research*, vol. 4, 2015.

[4] T. Gaudelet, N. Malod-Dognin, and N. Pržulj, "Higher-order molecular organization as a source of biological function," *Bioinformatics*, vol. 34, no. 17, pp. i944–i953, 2018.

[5] K. A. Shastry and H. Sanjay, "Machine learning for bioinformatics," in *Statistical Modelling and Machine Learning Principles for Bioinformatics Techniques, Tools, and Applications*. Springer, 2020, pp. 25–39.

[6] E. Pérez, O. Reyes, and S. Ventura, "Convolutional neural networks for the automatic diagnosis of melanoma: An extensive experimental study," *Medical image analysis*, vol. 67, p. 101858, 2020.

[7] V. Chaurasia and S. Pal, "A novel approach for breast cancer detection using data mining techniques," *International Journal of Innovative Research in Computer and Communication Engineering (An ISO 3297: 2007 Certified Organization) Vol*, vol. 2, 2017.

[8] K. Tomczak, P. Czerwińska, and M. Wiznerowicz, "The cancer genome atlas (tcga): an immeasurable source of knowledge," *Contemporary oncology*, vol. 19, no. 1A, p. A68, 2015.

[9] S. Ventura and J. M. Luna, *Supervised Descriptive Pattern Mining*. Springer, 2018.

[10] F. Padillo, J. M. Luna, and S. Ventura, "LAC: library for associative classification," *Knowledge Based Systems*, vol. 193, p. 105432, 2020.

[11] R. Vengateshkumar, S. Alagukumar, and R. Lawrance, "Analysis of microarray gene expression data using boolean association rule mining," *Int. J. Innov. Technol. Creat. Eng*, vol. 7, no. 5, pp. 412–416, 2017.

[12] A.-L. Boulesteix, G. Tutz, and K. Strimmer, "A cart-based approach to discover emerging patterns in microarray data," *Bioinformatics*, vol. 19, no. 18, pp. 2465–2472, 2003.

[13] J. Li and L. Wong, "Identifying good diagnostic gene groups from gene expression profiles using the concept of emerging patterns," *Bioinformatics*, vol. 18, no. 5, pp. 725–734, 2002.

[14] R. Vengateshkumar, S. Alagukumar, and R. Lawrance, "Boolean association rule mining on microarray gene expression data," in *Advanced Computing and Intelligent Engineering*. Springer, 2020, pp. 99–111.

[15] M. E. Ritchie, B. Phipson, D. Wu, Y. Hu, C. W. Law, W. Shi, and G. K. Smyth, "limma powers differential expression analyses for rna-sequencing and microarray studies," *Nucleic acids research*, vol. 43, no. 7, pp. e47–e47, 2015.

[16] M. D. Robinson, D. J. McCarthy, and G. K. Smyth, "edgeR: a Bioconductor package for differential expression analysis of digital gene expression data," *Bioinformatics*, vol. 26, no. 1, pp. 139–140, 11 2009.

[17] T. Uno, M. Kiyomi, and H. Arimura, "Lcm ver. 3: collaboration of array, bitmap and prefix tree for frequent itemset mining," in *Proceedings of the 1st international workshop on open source data mining: frequent pattern mining implementations*, 2005, pp. 77–86.

[18] J. Kang, C. M. Sergio, R. L. Sutherland, and E. A. Musgrove, "Targeting cyclin-dependent kinase 1 (cdk1) but not cdk4/6 or cdk2 is selectively lethal to myc-dependent human breast cancer cells," *BMC Cancer*, vol. 14, no. 1, p. 32, Jan 2014.

[19] B. Yu, Y. Ding, X. Liao, C. Wang, B. Wang, and X. Chen, "Overexpression of parpbp correlates with tumor progression and poor prognosis in hepatocellular carcinoma," *Digestive Diseases and Sciences*, Apr 2019.

[20] S. Winslow, K. E. Lindquist, A. Edsjö, and C. Larsson, "The expression pattern of matrix-producing tumor stroma is of prognostic importance in breast cancer," *BMC cancer*, vol. 16, no. 1, p. 841, 2016.

[21] M. Mura, R. Swain, X. Zhuang, H. Vorschmitt, G. Reynolds, S. Durant, J. Beesley, J. Herbert, H. Sheldon, M. Andre *et al.*, "Identification and angiogenic role of the novel tumor endothelial marker clec14a," *Oncogene*, vol. 31, no. 3, p. 293, 2012.

[22] P. Bi, F. Yue, A. Karki, B. Castro, S. E. Wirbisky, C. Wang, A. Durkes, B. D. Elzey, O. M. Andrisani, C. A. Bidwell, J. L. Freeman, S. F. Konieczny, and S. Kuang, "Notch activation drives adipocyte dedifferentiation and tumorigenic transformation in mice," *Journal of Experimental Medicine*, vol. 213, no. 10, pp. 2019–2037, 2016.

[23] F. C. M. Zoppino, M. E. Guerrero-Gimenez, G. N. Castro, and D. R. Ciocca, "Comprehensive transcriptomic analysis of heat shock proteins in the molecular subtypes of human breast cancer," *BMC Cancer*, vol. 18, no. 1, p. 700, Jun 2018.

Aplicación de técnicas de Aprendizaje Automático para el análisis de datos de acciones de juego en voleibol

Francisco Aragón-Royón[1], Elia Mercado-Palomino[2], Aurelio Ureña Espa[2], José Manuel Benítez[1]
[1] *Dept. de Ciencias de la Computación e Inteligencía Artificial, DiCITS, DaSCI, iMUDS*
[2] *Dept. de Educación Física y Deportiva, iMUDS*
Universidad de Granada. 18071, Granada, España
far@decsai.ugr.es

***Resumen*—En el deporte de alta competición el análisis tanto del riesgo de lesión como de los factores de rendimiento son dos aspectos de vital importancia. En voleibol, existe una carencia en la investigación de indicadores de rendimiento, y aunque se han estudiado los factores de riesgo de lesión, las situaciones de estudio están aún lejos de la realidad del juego. La inteligencia computacional aplicada a conjuntos de datos experimentales puede ayudar a identificar conocimiento relevante sobre el problema. Tras un riguroso diseño experimental, en este trabajo describimos la aplicación de distintas técnicas de análisis de datos para identificar las articulaciones más relevantes y determinar si existen diferencias en el desempeño de las extremidades inferiores en una acción real de juego de voleibol, el bloqueo. Los resultados muestran que se puede diferenciar entre extremidades e identifican una serie de variables relevantes que pueden ser tenidas en cuenta para la mejora del rendimiento y la reducción del riesgo de lesión.**

***Palabras Clave*—Voleibol, Riesgo lesivo, Indicador rendimiento, Extremidades inferiores, Aprendizaje Automático, Selección de Características**

I. Introducción

Uno de los objetivos principales en el deporte de alta competición es conseguir el máximo rendimiento. Para ello es imprescindible proteger al deportista y prevenir los posibles efectos lesivos de la alta repetición de habilidades específicas. En un deporte concreto como el voleibol, tareas específicas como saltar, bloquear o aterrizar tras un salto, deben realizarse junto con movimientos direccionales rápidos lo que propicia que los jugadores corran el riesgo de sufrir lesiones. La cadera, rodilla y tobillo han sido señaladas como las articulaciones más frecuentemente lesionadas en voleibol [15], y por ello su riesgo lesivo ha sido estudiado en los aterrizajes de los saltos [25] [3]. Pero los protocolos que han empleado hasta ahora para su estudio están lejos de situaciones reales de juego. Por otro lado, en el ámbito deportivo no existe aún un consenso claro sobre si esto afecta de igual manera a las extremidades inferiores. En algunos casos se ha planteado la simetría entre extremidades [18], pero otros estudios contradicen este supuesto [11]. Algunos trabajos como [16] han estudiado la existencia de diferencias entre extremidades en los aterrizajes de voleibol, pero se necesitan estudios más exahustivos. Por estas razones, la prevención de los riesgos específicos de lesión puede estar siendo tratada de manera limitada al no tener un conocimiento claro de los factores más influyentes. Identificar correctamente estos factores y todas sus interrelaciones en situaciones reales de juego se torna en una tarea de vital importancia.

La Inteligencia Computacional puede ayudar a estudiar en profundidad estas cuestiones. La aplicación de técnicas de Inteligencia Artificial, y más concretamente de Aprendizaje Automático, como herramientas transversales está expandiéndose en el ámbito de las Ciencias del Deporte y está generando un gran número de estudios analíticos [7]. Estos estudios abren la posibilidad de obtener un conocimiento adicional de los factores más influyentes en el desempeño de la actividad deportiva que al ser estudiados pueden ayudar a tomar decisiones adecuadas relacionadas con el rendimiento o la prevención de lesiones del deportista. Estas técnicas son capaces de analizar grandes conjuntos de datos y de aprender automáticamente relaciones y patrones complejos inherentes que son desconocidos incluso para los expertos. Todo esto es también posible gracias a la tecnología que permite obtener una gran cantidad de datos de los jugadores que hasta ahora no se estaban teniendo en cuenta ni se estaban recogiendo.

En este artículo proponemos el uso de métodos de Aprendizaje Automático para realizar una clasificación de la condición de las extremidades en una serie de saltos realizados por jugadoras de voleibol. Estos saltos se han realizado en un escenario experimental basado en una situación real de bloqueo, proporcionando una similitud en el aterrizaje lo más cercana posible a la realidad del juego. La propuesta está basada en el ajuste de dos modelos de clasificación mediante técnicas XGBoost [5] y Extreme Learning Machines (ELM) [10] con la finalidad de determinar si se presentan diferencias significativas entre el desempeño de ambas piernas en los diferentes saltos realizados. El estudio se completa con el uso de una selección de características (SC) para identificar las variables más relevantes.

La metodología propuesta ha sido aplicada en varias situaciones planteadas por expertos. Estas situaciones representan saltos hacia direcciones de dominancia y no dominancia, y se han estudiado las diferencias entre la pierna dominante y la no dominante, y la pierna exterior e interior. Los resultados

muestran que los métodos de Aprendizaje Automático son capaces de diferenciar las extremidades durante los saltos, y por lo tanto existen diferencias significativas entre ellas. Lo que debe de ser tenido en cuenta a la hora de estudiar los riesgos de lesión y el rendimiento en las acciones de juego. Esto es así, tanto si se emplean todas las características disponibles como si se seleccionan un subconjunto de características prometedoras. Aunque mediante la selección de características se deduce que existe un subconjunto de variables referentes a las articulaciones con una mayor incidencia en la realización de los movimientos planteados.

II. Trabajos publicados sobre la temática

La cantidad de trabajos en el ámbito deportivo que emplean técnicas de Aprendizaje Automático es cada vez mayor. En [6] hay una revisión sistemática de muchos de estos trabajos, que de manera general, se puede categorizar en dos grandes grupos de trabajos. Un primer grupo que estudia los indicadores que influyen en el rendimiento, y un segundo grupo de trabajos que abordan los aspectos lesivos.

Con respecto a los primeros, los trabajos analizan los datos de deportistas para encontrar los indicadores de rendimiento que presentan una mayor influencia en deportes como ciclismo [21], natación [9], fútbol [8] o biatlón [17]. Posteriormente suelen realizar una predicción de resultados en base a dichos indicadores. Estos trabajos emplean en gran medida redes neuronales, pero también se han empleado modelos basados en árboles, redes bayesianas o técnicas de aprendizaje no supervisado como *k-means*. En esta misma línea, algunos trabajos tratan de reducir la dimensión del problema y seleccionar las mejores características. En [20] se realiza una extracción de características previa a la aplicación una red neuronal para evaluar los ejercicios realizados en el entrenamiento con pesas. Mientras que en [23] se realiza una selección de características basada en métodos filtro y wrapper para predecir el éxito futuro de jugadores de tenis. El objetivo de todos estos trabajos centrados en el rendimiento reside en determinar información valiosa para el proceso de toma de decisiones y así asistir a los expertos para desarrollar mejores programas de entrenamiento, de estrategias o de selección de deportistas.

Por otro lado, con respecto a los trabajos que abordan el estudio de las lesiones, se ha abordado la clasificación [19] y la predicción [12] de lesiones de la rodilla mediante técnicas de Aprendizaje Automático como SVM y aprendizaje profundo. Adicionalmente, en [14] se analiza si mediante la detección de acciones de juego con técnicas de Aprendizaje Automático y Deep Learning se pueden identificar factores de riesgo en volley playa. En [24] y [13] se investiga la predicción de lesiones en fútbol, empleando para ello enfoques basados en métodos de Aprendizaje Automático. En ambos trabajos se realiza una selección de características previa con la finalidad de determinar las características más relevantes.

No existen trabajos aún que determinen indicadores de rendimiento o de riesgo de lesión en voleibol mediante técnicas de Inteligencia Computacional.

III. Metodología

En esta sección se formaliza el problema que se aborda en este artículo, se presenta el conjunto de datos y posteriormente se detalla nuestra propuesta para el análisis de los datos.

III-A. Análisis del problema y formulación

El estudio parte de un experimento realizado sobre un conjunto de jugadoras semi-profesionales de voleibol. Consiste en simular una situación real de juego en la que se realiza un salto lateral direccional para bloquear y en su aterrizaje. Para que la mecánica sea lo más cercana a una situación real se han respetado las direcciones, distancia y altura de los saltos tal y como se realizan en la competición. Con el experimento se obtienen mediciones de varias articulaciones de las jugadoras en los diferentes planos y las fuerzas ejercidas por estas en las caídas de los saltos. Esto se consigue mediante sistemas de captura de movimiento, posicionamiento 3D, y cálculo de la cinemática tridimensional de la cadera, rodilla y tobillo.

Cada uno de los saltos ha sido categorizado en función de la dirección del salto y además se ha asignado un rol a cada una de las extremidades. La dirección del salto puede ser hacia el lado dominante o hacia el lado no dominante. La dominancia de salto es establecida como la secuencia natural de pasos empleados al realizar un salto de remate en voleibol. Un salto dominante se realiza hacia el lado contrario a la pierna dominante, mientras que un salto no dominante se realiza hacia el lado de la pierna dominante. El papel de las extremidades viene determinado por: a) la dominancia, siendo la pierna hábil la pierna dominante y la pierna potente la pierna no dominante, y b) la posición de las extremidades durante el aterrizaje, siendo la pierna exterior la más alejada en el aterrizaje al punto de inicio del salto y la pierna interior la más cercana al punto de partida del salto. Luego en función de la dirección del salto, las extremidades dominantes y no dominantes pueden tener un rol de posición distinto.

Con este punto de partida, en este trabajo se pretende determinar si existen diferencias significativas entre el comportamiento de las extremidades inferiores de las jugadoras. Además, se pretende descubrir aquellas variables de las extremidades que presenten una mayor incidencia en el desempeño de los saltos, de forma que sirvan como indicadores que puedan ser usados en futuros análisis de rendimiento y del riesgo de lesión. Para alcanzarlo, el problema se aborda empleando diferentes técnicas de Aprendizaje Automático.

Los expertos han determinado que las cuatro preguntas a investigar más relevantes son las siguientes:

- Pregunta 1.1. ¿Existen diferencias significativas entre extremidad dominante (interior) y no dominante (exterior) en saltos hacia dirección dominante?. En este caso se pretende contrastar si existe diferencia entre extremidades cuando el salto es hacia el lado dominante. El estudio se realiza sobre los saltos dominantes en la extremidad exterior / no dominante, y los saltos dominantes en la extremidad interior / dominante. Esta equivalencia es así ya que para los saltos dominantes la extremidad

exterior siempre va a ser la extremidad no dominante, y la extremidad interior siempre va a ser la extremidad dominante.

- Pregunta 1.2. ¿Existen diferencias significativas entre extremidad dominante (exterior) y no dominante (interior) en saltos hacia dirección no dominante? En este caso se pretende contrastar si existe diferencia entre extremidades cuando el salto es hacia el lado no dominante. El estudio se realiza sobre los saltos no dominantes en la extremidad exterior / dominante, y en los saltos no dominantes en la extremidad interior / no dominante. Esta equivalencia es así ya que para los saltos no dominantes la extremidad exterior siempre va a ser la extremidad dominante, y la extremidad interior siempre va a ser la extremidad no dominante.
- Pregunta 2.1. ¿Existen diferencias significativas entre la extremidad exterior cuando esta va hacia el lado dominante y cuando va hacia el lado no dominante? En este caso se pretende contrastar si existe diferencia en la extremidad exterior cuando el salto es hacia el lado dominante o hacia el lado no dominante. El estudio se realiza sobre los saltos dominantes en la extremidad exterior, y en los saltos no dominantes en la extremidad exterior.
- Pregunta 2.2. ¿Existen diferencias significativas entre la extremidad interior cuando esta va hacia el lado dominante y cuando va hacia el lado no dominante? En este caso se pretende contrastar si existe diferencia en la extremidad interior cuando el salto es hacia el lado dominante o hacia el lado no dominante. El estudio se realiza sobre los saltos dominantes en la extremidad interior, y en los saltos no dominantes en la extremidad interior.

Una forma efectiva de responder a estas preguntas es formulándolas en términos de problemas de clasificación. Es decir, consideramos los datos obtenidos en la experimentación como datos de entrada y cada una de las extremidades como valor de salida. Es un problema de asociación entre entradas y salidas, y dado el número pequeño de posibles valores de la salida (dos en todos los casos), es un problema de clasificación, no un problema de regresión. Por tanto, el Aprendizaje Automático puede ayudar a obtener conocimiento en este caso mediante la aplicación de técnicas de clasificación.

III-B. Descripción del conjunto de datos

Como se ha mencionado en la sección III-A, los datos se obtienen de un experimento basado en una situación real de bloqueo y aterrizaje del salto de catorce jugadores de voleibol semi profesionales. El número total de todos los saltos realizados por las jugadoras es de 376 saltos. Para cada uno de los saltos de cada jugadora se obtiene el valor de 32 variables, que son medidas en el momento denominado como F1. El momento F1 es el instante de pico de fuerza medido en la plataforma de caída de cada pierna, que es cuando suelen ocurrir con mayor frecuencia las lesiones [22]. Por lo tanto, por cada salto de cada jugadora se generan dos filas, cada una con los valores de las 32 variables para cada pierna. Así pues, la matriz de datos está compuesta por 32 columnas y 752 filas correspondientes a los valores de las variables en cada una de las piernas en los saltos de las jugadoras (376 saltos por dos piernas). Las 32 variables que se han tomado en consideración para este estudio son: 9 variables correspondientes a los ángulos (grados) en el plano coronal, sagital y transversal tanto para la cadera como para la rodilla y el tobillo; 9 variables correspondientes a la velocidad angular (grados/s) en los mismos planos y articulaciones; 9 variables referentes al momento de fuerza (Nm/kg) en igual condiciones; 3 variables en el plano sagital para la absorción de fuerza del ligamento (J/kg); y otras 2 variables correspondientes a la fuerza ejercida verticalmente en el suelo (N) y la carga (N/s) en el plano transversal. Las etiquetas correspondientes a cada una de las filas se generan en función de a qué pierna se corresponden los valores (no dominante-dominante y exterior-interior) y según si el salto es hacia el lado dominante de la jugadora o no.

III-B1. Conjunto de datos de estudio: Para cada uno de las preguntas de investigación planteadas el conjunto de datos de partida es el mismo. Sin embargo, en cada caso se empleará un subconjunto diferente en función de aquellas filas que se necesiten para resolver el problema planteado. Así pues, tenemos que para cada pregunta el conjunto de datos queda de la siguiente manera: a) Pregunta 1.1. Se tienen en cuenta las filas en las que las variables se corresponden con saltos hacia el lado dominante (etiquetadas según correspondan a la pierna interior (dominante) o exterior (no dominante)). b) Pregunta 1.2. Filas en las que las variables se corresponden con saltos hacia el lado no dominante (etiquetadas según los valores del salto correspondan a la pierna interior (no dominante) o exterior (dominante)). c) Pregunta 2.1. Filas en las que las variables se corresponden con la pierna exterior (etiquetadas según el salto haya sido hacia el lado dominante o hacia el lado no dominante). d) Pregunta 2.2. Filas en las que las variables se corresponden con la pierna interior (etiquetadas según el salto haya sido hacia el lado dominante o hacia el lado no dominante).

III-C. Proceso de selección de características

La selección de características (SC) es una técnica de preprocesamiento. SC se define como el proceso de selección de un subconjunto de características relevantes de cara al estudio (p.ej. para la construcción de un modelo predictivo). Por lo que el objetivo es reducir la dimensionalidad de los datos quedándose con las características que tienen mayor influencia en la salida. La selección de características pueden ser vista como una búsqueda de un subconjunto de características óptimo en el espacio de todas las características. Luego la SC se puede entender como un proceso formado por una estrategia de búsqueda dentro del espacio de características (2^n, donde n es el número de características) guiada por una evaluación de los subconjuntos [2]. Por lo tanto, el proceso de selección de características está formado por un método de búsqueda y método de evaluación. Una de las clasificaciones

más aceptadas de los métodos de selección de características viene dada por la forma de aplicación. Esto está directamente relacionado con la manera de evaluación de los subconjuntos de características. En este trabajo se han considerado métodos basados en dos enfoques: filtro y wrapper. a) Métodos de filtro: estos métodos emplean una medida de la utilidad para evaluar la idoneidad de los subconjuntos de características. b) Métodos de envoltorio (*wrapper*): estos métodos utilizan un algoritmo de aprendizaje para evaluar la calidad de los subconjuntos de características.

III-D. Metodología propuesta

Para dar respuesta a las preguntas planteadas hemos diseñado un proceso de análisis de datos basado en en la construcción de clasificadores. Concretamente para los casos 1.1 y 1.2 se pretende determinar si los datos se corresponden a la extremidad interior o a la exterior, mientras que para los casos 2.1 y 2.2 se pretende determinar si los datos corresponden a un salto hacia el lado dominante o hacia el lado no dominante. Además, también proponemos la aplicación de un proceso de selección de características previo a la construcción de los modelos de clasificación, con el objetivo de mejorar el rendimiento de los modelos e identificar las variables que tienen una mayor influencia en cada caso. El enfoque de cada procedimiento se detalla a continuación.

III-D1. Clasificadores usando todas las características: Esta propuesta consiste en ajustar los modelos de clasificación empleando todas las características disponibles. El procedimiento consta de los siguientes pasos:

1. Preprocesamiento. El conjunto de datos se particiona mediante muestreo aleatorio en un subconjunto de entrenamiento y otro de test. Los datos se normalizan mediante un centrado y un escalado.
2. Ajuste de hiperparámetros. Como métodos de construcción de clasificadores se han fijado XGBoost y ELM. Se determinan los valores más adecuados de los hiperparámetros de las técnicas. Para hacer esto, se genera una rejilla con los valores de los parámetros y con cada una de las combinaciones de la rejilla se realiza una validación cruzada de factor k sobre los datos de la partición de entrenamiento. La combinación de hiperparámetros que obtenga mejor valor en su subconjunto de prueba es la que se establece como la mejor. Estos parámetros han sido los utilizados para la construcción de los clasificadores.
3. Construcción de los modelos. Se emplean los métodos XGBoost y ELM con los hiperparámetros seleccionados en el paso previo para construir los clasificadores a partir de los datos de entrenamiento.
4. Prueba del modelo. Los modelos entrenados se utilizan para clasificar el subconjunto de datos de test, y obtener una evaluación de los clasificadores ajustados. Como forma de medir el rendimiento de los modelos se emplea la precisión (*accuracy*) (ACC). La ACC se calcula como la proporción de instancias clasificadas de manera correcta entre el total de instancias de test.

$$\text{ACC} = \frac{|correctas - totales|}{totales} \cdot 100 \qquad (1)$$

III-D2. Clasificadores con selección de características: Esta propuesta consiste en aplicar previamente una selección de características para seleccionar un subconjunto de características reducido y más representativo. Posteriormente, los datos de entrenamiento y test son proyectados según las características seleccionadas. El procedimiento consta de los siguientes pasos:

1. Preprocesamiento. El conjunto de datos se particiona en un subconjunto de entrenamiento y otro de test. Los datos se normalizan mediante un centrado y un escalado.
2. Selección de características. Se realiza una selección de características sobre los datos de entrenamiento. Este proceso realiza con validación cruzada. Cada combinación anterior de búsqueda+(filtro/wrapper) proporciona el subconjunto de características seleccionado.
3. Ajuste de hiperparámetros. Se determinan los valores más efectivos de los hiperparámetros para las técnicas de construcción de clasificadores usando una rejilla y validación cruzada.
4. Construcción de los modelos. Se ajusta un modelo de clasificación sobre los datos de entrenamiento con cada uno de los subconjuntos de características obtenidos en el proceso de SC anterior. Los clasificadores se crean mediante métodos XGBoost y ELM usando los mejores parámetros obtenidos en el procedimiento anterior.
5. Test de los modelos. Los modelos entrenados se prueban sobre el subconjunto de datos de test. Esto permite obtener una evaluación de la capacidad de generalización de los clasificadores. La métrica de evaluación del rendimiento es la ACC descrito en la expresión 1.

IV. Diseño experimental

Para evaluar la eficacia de la metodología de análisis propuesta, la hemos implementado usando el lenguaje de programación R y la hemos aplicado al conjunto de datos descrito en la sección III-B. Hemos utilizado los siguientes paquetes de R y aplicado los siguientes valores de hiperparámetros:

- Partición del conjunto de datos y validación cruzada. El conjunto completo de datos se ha particionado mediante muestreo aleatorio en entrenamiento, 80 %, y prueba, 20 %. Para la validación cruzado se han usado 5 dobleces (k = 5).
- Construcción de los clasificadores.
  a) XGBoost: Paquete `caret` [4]. Se ha usado un modelo extreme gradient boosting de la biblioteca `xgboost`. Los parámetros se han dejado por defecto, salvo tres de ellos que se han ajustado: 1) $nround$, con valores 5, 10, 25, 100, 2) $max.depth$, con valores 2, 3, 5 y 3) eta con valores 0.1, 0.2, 0.3, 0.4.
  b) Extreme Machine Learning: Paquete `caret`. Se ha empleado una red con una única capa oculta, *single hidden-layer feedforward neural network* (SLFN) de

la biblioteca `elmNN`. Los parámetros se han dejado por defecto salvo: 1) $nhid$, con valores 5, 10, 25, 100 y 2) $actfun$, con valores sig y $relu$.

- Selección de características. Paquete `FSinR`[1] [1]. El proceso de SC se descompone en un algoritmo de búsqueda y un método de evaluación. Como métodos de búsqueda se han empleado: a) Ant Colony Optimization (ACO) con parámetros: $population$=10, $iter$=20, q=5, p=0.15, b=0.5, $tmax$=0.5, $tmin$=0.1 y $mode$=3 para los métodos de wrapper, para filtro cambian b=0 y $t0$ = 0.30. b) Genetic Algorithm (GA) con parámetros $popSize$ = 25, $pcrossover$ = 0.8, $pmutation$ = 0.1 y $maxiter$=20 tanto para los métodos de filtro como de wrapper. c) Hill-Climbing (HC) con parámetros por defecto. d) Las Vegas (LV) con el parámetro K=50. e) Tabu Search (TS) con parámetros: $tamTabuList$ = 6, $iter$ = 25, $intensification$=1, $iterIntensification$ = 15, $diversification$ = 1, $iterDiversification$=15. Por otro lado, como métodos de filtro se han usado determinationCoefficient y RFSM con los parámetros por defecto. Y los métodos de wrapper empleados han sido XGBoost y ELM. Para estos métodos de wrapper se ha vuelto a emplear el paquete `caret`.

V. Resultados experimentales

En esta sección se presentan los resultados obtenidos en los experimentos y un análisis y discusión de los mismos.

La Tabla I muestra los resultados obtenidos en la experimentación. La primera columna muestra el caso de estudio al que pertenecen los valores de cada fila. Las cuatro columnas siguientes muestran la precisión obtenida en el conjunto de prueba con los dos clasificadores, distinguiendo si se ha usado SC o no. Se observa que en cualquier combinación de clasificado y uso de SC o no se obtienen resultados muy buenos, Para todas las preguntas de investigación existe al menos una combinación que obtiene resultados iguales o superiores al 96 %, lo que significa que los resultados pueden calificarse de excelentes. En todos los casos, se pueden encontrar clasificadores capaces de discriminar qué al extremidad o dirección del salto. Es decir, se detectan diferencias entre la extremidad interior y exterior en los casos 1.1 y 1.2, y entre los saltos hacia el lado dominante y no dominante en los casos 2.1 y 2.2.

Se detecta una ligera superioridad en los resultados que buscan diferencias entre piernas (casos 1.1 y 1.2) que en los casos en los que se pretende distinguir la dirección del salto (casos 2.1 y 2.2). Sin embargo, el factor más destacado es la aplicación de selección de características: queda claro que el resultado es mejor cuando se aplica SC, pues los resultados de las dos últimas columnas son mejores que los de las respectivas columnas previas. En los resultados no se muestra con detalle mediante qué métodos de búsqueda y de filtro/wrapper ha obtenido la SC los subconjuntos de características con los que se han ajustado estos mejores modelos. Para aclarar esto, se reseña que los subconjuntos óptimos se han obtenidos por métodos de filtro o de wrapper en proporción similar, así como con prácticamente todos los métodos de búsqueda empleados.

[1]https://cran.r-project.org/package=FSinR

Cuadro I
ACC obtenido por los modelos con todas las características y con una Selección de Características

Pregunta	ACC todas carac.		ACC selección carac.	
	XGBoost	ELM	XGBoost	ELM
1.1	97.297	97.297	**98.649**	**98.649**
1.2	90.789	97.368	97.368	**98.684**
2.1	94.667	93.333	93.333	**96.000**
2.2	94.667	92.000	96.000	**97.333**

La Tabla II muestra las características que han sido seleccionadas en un mayor número de subconjuntos en los resultados del proceso de selección de características. La primera columna hace referencia al caso de estudio, la segunda columna incluye las características más relevantes, mientras que en la última columna se muestra su frecuencia de selección (se muestran aquellas características que han alcanzado un porcentaje de selección del 75 % o mayor). Se puede apreciar como cada caso tiene su propio conjunto de características más usadas, pero en los casos que hacen referencia a la dirección del salto, 2.1 y 2.2, la variable Ankle Mom Z coincide como la variable más empleada con un alto porcentaje de aparición. La importancia de esta variable queda latente por tanto a la hora de determinar la dirección del salto. También se puede apreciar como no hay ninguna característica del eje X entre las más seleccionadas por el proceso de SC.

Cuadro II
Características con un mayor porcentaje de selección para cada caso

Pregunta	Características	%
1.1	Ankle Mom Z	85
	Hip Angle Y	85
	Ankle Mom Y	80
	Knee Mom Y	80
	Ankle Mom X	75
	Knee Angle Y	75
1.2	Knee Vel_Ang Z	85
	Knee Angle Z	75
	FP Z	75
	Ankle Mom Y	75
2.1	Ankle Mom Z	95
	Ankle Mom X	80
	Ankle Mom Y	80
	Knee Mom Z	75
2.2	Ankle Mom Z	95
	Ankle Mom Y	80
	Ankle Vel_Ang Y	80
	Knee Vel_Ang Y	80
	Knee Mom Y	75
	Knee Angle Y	75

Estos resultados ponen de manifiesto la importancia de tener en cuenta la extremidad y la dirección en aterrizajes en voleibol. Las variables biomecánicas medidas han demostrado que las extremidades tienen un desempeño diferente según la dirección del salto. Por lo tanto, estas variables pueden ser

útiles como indicadores de rendimiento y ofrecer información valiosa a los entrenadores con la que estudiar mejoras en el rendimiento de los deportistas, así como ayudar a los equipos médicos a evaluar el riesgo de lesión en la acción.

VI. Conclusión

En este trabajo se han aplicado técnicas de Aprendizaje Automático para el análisis de datos de acciones deportivas en voleibol. Más en concreto, se han estudiado cuatro preguntas definidas por expertos relacionadas con la efectividad de ciertas acciones de juego y el riesgo de lesión asociado. Las preguntas se centran en determinar la existencia de diferencias significativas entre ambas extremidades durante la realización de bloqueos en voleibol. Este problema de discriminación se ha formulado como un problema de clasificación y se ha diseñado una metodología sencilla para abordarlo. La metodología incluye el uso de distintas técnicas de construcción de clasificadores y el empleo de procedimientos de selección de características.

La metodología se ha aplicado para el análisis de un conjunto de datos obtenidos de la simulación en laboratorio de situaciones de juego real. Los resultados muestran que existen diferencias significativas entre el comportamiento de ambas extremidades cuando se realiza un salto tanto en dirección dominante como en dirección no dominante. Además también se han encontrado diferencias entre una misma extremidad en función de la dirección del salto. Mientras que por otro lado, la aplicación de un proceso de SC nos ha ayudado para descubrir cuáles son las características que presentan un mayor impacto en los modelos de clasificación. La importancia del conocimiento extraído valida la aplicación de técnicas de Aprendizaje Automático en problemas del ámbito de la actividad física y el deporte.

Las conclusiones derivadas de este análisis proporcionan información que ayuda a comprender cómo actúan las extremidades inferiores durante la acción de salto-aterrizaje de un bloqueo en voleibol. Destaca la importancia de tener en consideración la dirección del salto y el papel de las extremidades, y la identificación de qué características tienen más incidencia en el problema. Por tanto, es interesante que sean consideradas en estudios futuros sobre el rendimiento del deportista o el riesgo de lesión en voleibol.

Agradecimientos

Esta investigación ha sido financiada parcialmente por FEDER/Junta de Andalucía, Consejería de Economía y Conocimiento, proyectos A.TIC.388.UGR18 y P18-TP-5168.

Referencias

[1] Aragón-Royón, F., Jiménez-Vílchez, A., Arauzo-Azofra, A., Benítez, J.M. *FSinR: an exhaustive package for feature selection*, arXiv, 2020. URL: https://arxiv.org/abs/2002.10330

[2] Arauzo-Azofra, A., Aznarte, J.L., Benítez, J.M. *Empirical study of feature selection methods based on individual feature evaluation for classification problems*, Expert Systems with Applications, 38(7) 8170-8177, 2011

[3] Avedesian, J.M., Judge, L.W., Wang, H., Dickin, D.C. *The biomechanical effect of warm-up stretching strategies on landing mechanics in female volleyball athletes*, Sports Biomechanics, 1-14, 2018

[4] Kuhn, M. *Building Predictive Models in R Using the caret Package*, Journal of Statistical Software, 28(5) 1-26, 2016

[5] Chen, T., Guestrin, C. *XGBoost: A Scalable Tree Boosting System*, Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 785–794, 2008

[6] Claudino, J.G., Capanema, D., de Souza, T.V., Serrao, J.C., Pereira, A., Nassis, G.P. *Current Approaches to the Use of Artificial Intelligence for Injury Risk Assessment and Performance Prediction in Team Sports: a Systematic Review*, Sports Medicine, 5(28), 2019

[7] Cust, E.E., Sweeting, A.J., Ball, K., Robertson, S. *Machine and deep learning for sport-specific movement recognition: a systematic review of model development and performance*, Journal of Sports Sciences, 37(5), 568-600, 2019

[8] Dutt-Mazumder, A., Button, C., Robins, A., Bartlett, R. *Neural Network Modelling and Dynamical System Theory*, Sports Med, 41, 1003–1017, 2011

[9] Hedelmann-Nusser, J., Hohmann, A., Henneberg, B.*Modeling and prediction of competitive performance in swimming upon neural networks*, European Journal of Sport Science, 2(2), 1-10, 2010

[10] Huang, G. Zhu, Q., Siew, C. *Extreme learning machine: A new learning scheme of feedforward neural networks*, Proceedings of the International Joint Conference on Neural Networks, 985–990, 2006

[11] Ithurburn, M.P., Paterno, M.V., Ford, K.R., Hewett, T.E., Schmitt, L.C. *Young Athletes With Quadriceps Femoris Strength Asymmetry at Return to Sport After Anterior Cruciate Ligament Reconstruction Demonstrate Asymmetric Single-Leg Drop-Landing Mechanics*, The American Journal of Sports Medicine, 43(11), 2727-2737, 2015

[12] Johnson, W.R., Mian, A., Lloyd, D.G., Alderson, J.A.*On-field player workload exposure and knee injury risk monitoring via deep learning*, Journal of Biomechanics, 93, 185-193, 2019

[13] Kampakis, S. *Predictive modelling of football injuries*, arXiv, 2016

[14] Kautz, T., Groh, B.H., Hannink, U.J., Strubberg J., Eskofier, B.M. *Activity recognition in beach volleyball using a Deep Convolutional Neural Network*, Data Min Knowl Disc, 31, 1678–1705, 2017

[15] Kilic, O., Maas, M., Verhagen, E., Zwerver, J., Gouttebarge, V. *Incidence, aetiology and prevention of musculoskeletal injuries in volleyball: A systematic review of the literature*, European Journal of Sport Science, 17(6) 765-793, 2017

[16] Marquez, W., Masumura, M.o., Ae, M. *The effects of jumping distance on the landing mechanics after a volleyball spike*, Sports biomechanics, 8, 154-66, 2009

[17] Maier, T., Meister, D., Trösch, S., Wehrlin, J.P. *Predicting biathlon shooting performance using machine learning*, Journal of Sports Sciences, 36(20), 2333-2339, 2018

[18] McPherson, A.L., Dowling, B., Tubbs, T.G., Paci, J.M. *Modelling and analysing track cycling Omnium performances using statistical and machine learning techniques*, Journal of Sports Sciences, 31(9), 954-962, 2013

[19] Mazlan, S.S., Ayob, M.Z., Kadir Bakti, Z.A. *Anterior cruciate ligament (ACL) injury classification system using support vector machine (SVM)*, 2017 International Conference on Engineering Technology and Technopreneurship (ICE2T), 1-5, 2017

[20] Novatchkov, H., Baca, A. *Artificial intelligence in sports on the example of weight training*, Journal of sports science and medicine, 12(1), 27–37, 2013

[21] Ofoghi, B., Zeleznikow, J., Dwyer, D., Macmahon, C. *Sagittal plane kinematic differences between dominant and non-dominant legs in unilateral and bilateral jump landings*, European Journal of Sport Science, 22, 54 - 60, 2016

[22] Olsen, O.E., Myklebust, G., Engebretsen, L., Bahr, R. *Injury Mechanisms for Anterior Cruciate Ligament Injuries in Team Handball A Systematic Video Analysis*, The American journal of sports medicine, 32, 1002-1012, 2004

[23] Panjan, A., Sarabon, N., Filipcic, A. *Prediction of the successfulness of tennis players with machine learning methods*, Kinesiology, 42, 98-106, 2010

[24] Rossi, A., Pappalardo, L., Cintia, P., Iaia, F.M., Fernández, J., Medina, D. *Effective injury forecasting in soccer with GPS training data and machine learning*, PLOS ONE, 13(7), 1-15, 2018

[25] Zahradnik, D., Jandacka, D., Uchytil, J., Farana, R., Hamill, J. *Lower extremity mechanics during landing after a volleyball block as a risk factor for anterior cruciate ligament injury*, Physical Therapy in Sport, 16(1) 53 - 58, 2015

X Simposio de Teoría y Aplicaciones de Minería de Datos

Trabajos Destacados



Rotation Forest for Big Data*

Mario Juez-Gil, Álvar Arnaiz-González, Juan J. Rodríguez, Carlos López-Nozal, César García-Osorio
Dpto. Ingeniería Informática, Universidad de Burgos
Escuela Politécnica Superior, Avda. Cantabria s/n, 09006, Burgos, Spain

***Abstract*—This is a summary of our article published in Information Fusion [3], presented to the Multi-Conference CAEPIA'21 KeyWorks.**

The Rotation Forest classifier is a successful ensemble method for a wide variety of data mining applications. One of the drawbacks of Rotation Forest is the high training and prediction times, that makes it unfeasible for Big Data. This paper presents a novel MapReduce Rotation Forest design and its implementation under the Spark framework. Experimental results are obtained using different cloud-based cluster configurations. Bayesian tests are used to validate the method against two ensembles for Big Data: Random Forest and PCARDE classifiers. The proposed design incorporates the parallelization of both the PCA calculation and the tree training, providing a scalable solution that retains the performance of the original Rotation Forest and achieves a competitive execution time.

***Index Terms*—Rotation Forest, Random Forest, Ensemble Learning, Machine Learning, Big Data, Spark**

I. Summary

Ensemble methods have demonstrated their remarkable performance over the past few decades [4]. Their simplicity and flexibility make them useful in several domains. Moreover, their modularity (i.e., they are based on individual base classifiers or regressors) makes their parallelization feasible. Unfortunately, the number of ensemble algorithms available on Big Data frameworks is still limited. The aim of this paper is to present the MapReduce design and implementation of the well-known Rotation Forest ensemble [5] and its evaluation. To assess whether the Rotation Forest ensemble maintains its good performance in Big Data, it was thoroughly compared against the few ensemble algorithms available for Big Data within a Spark cluster by using several large data sets with a number of instances ranging between 400 000 and 11 000 000, and a number of attributes ranging between 11 and 2 000.

Despite the remarkable performance of Rotation Forest, its main drawback is that it is a time-consuming algorithm. A Rotation Forest classifier needs to perform multiple PCA calculations as one of its steps, and rotate both the training and the testing data, making it slower than other ensemble methods. Nonetheless, an efficient design and implementation in Big Data frameworks can make it suitable for the new challenges posed by the need to process large data sets.

We would like to thank the *Ministerio de Economía y Competitividad* of the Spanish Government for financing the project TIN2015-67534-P (MINECO/FEDER, UE) and the *Junta de Castilla y León* for financing the project BU055P20 (JCyL/FEDER, UE) both cofinanced from European Union FEDER funds. The research leading to these results has received funding from "La Caixa" Foundation and Caja Burgos Foundation, under agreement LCF/PR/PR18/51130007. This material is based upon work supported by Google Cloud.

The main contributions of the paper were:

- A novel MapReduce Rotation Forest design and its implementation under the Spark framework is presented.
- An extensive experimentation: Bayesian tests [1] are used to validate the method against two ensembles for Big Data (Random Forest and PCARDE classifiers).

Our proposal to compute PCA and to train the trees in parallel provides a scalable solution that achieves a competitive runtime, while retaining a remarkable performance level.

Training a Rotation Forest is a very time-consuming task, which is its main drawback, at least in the context of Big Data processing. This is mainly because PCA calculation requires more computing resources, which is greater as the number of instances and features increase. In Rotation Forest, data are transformed through a sparse rotation matrix computed by arranging K PCA rotation matrices (each of one calculated for K random feature subsets). Additionally, since Rotation Forest is an ensemble, as many sparse rotation matrices as the size of the ensemble will be computed (i.e., for an ensemble of size L, PCA is computed $L \times K$ times).

Nonetheless, it can be done in an efficient parallel way, as PCA can be solved using matrix algebra (i.e., singular value decomposition or covariance matrix calculations followed by an eigenvalue decomposition). In the Spark framework, PCA is already implemented using a parallel singular value decomposition (SVD) algorithm. Decision tree-based algorithms, such as Random Forest, can also be parallelized in many ways.

Both PCA and Random Forest implementations provided by Spark are used, in order to take advantage of Rotation Forest for Big Data processing.

The training stage of Rotation Forest is presented in Algorithm 1. The algorithm rotates the input data $\mathbf{X}$ and then trains a Random Forest of size T using the rotated data. The process is performed L times, in order to build an ensemble that is composed of L rotation matrices and L Random Forests. Hence, the total number of trees in the ensemble is $L \times T$.

The construction of the rotation matrix $\mathbf{R}^a$ is performed in lines 3 to 12. Initially, the feature set, $\mathbf{F}$, of the input data, $\mathbf{X}$, is randomly split into a partition, $\mathbf{Q}$, of K subsets (K is a parameter of the algorithm). For each feature subset, $\mathbf{S}$, of $\mathbf{Q}$, a submatrix, $\mathbf{W}$, is extracted from $\mathbf{X}$ that only contains the features in $\mathbf{S}$ (that is, $\mathbf{W}$ only contains a subset of columns of $\mathbf{X}$). The matrix, $\mathbf{W}$, is now further reduced by removing some of its rows, more specifically a random selection of classes is made and all instances not belonging to the selected classes are removed. The result is a new matrix, $\mathbf{W}'$. An additional reduction step generates the matrix, $\mathbf{W}''$,

Algorithm 1: Rotation Forest for Big Data.

Input: A training set $(\mathbf{X}, \mathbf{Y})$ where $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ defined in a feature set $\mathbf{F}$, $\mathbf{Y} = \{y_1, \ldots, y_n\}$ with labels $y_i \in \Omega = \{\omega_1, \ldots, \omega_c\}$ representing c classes, number of rotations L, number of trees T, number of feature subsets K, bootstrap size B.

Output: Trained ensemble E (tuples: rotation matrix $\mathbf{R}^a$, base classifier D).

```
E ← map i ∈ {1, ..., L}
    Q ← random partition of F into K subsets of features
    M ← map S ∈ Q
        W ← submatrix of X with the columns corresponding to the features in S
        Y' ← random selection of classes in Ω
        W' ← submatrix of W with rows corresponding to instances of classes in Y'
        W'' ← bootstrap sample of size B% of the number of instances in W'
        C ← rotation matrix from PCA(W'')
        emit ⟨C⟩
    R ← reduce (M)                    // block diagonal matrix
    P ← permutation matrix, matching the order of the features in F
    R^a ← P R
    D ← train-random-forest(X R^a, Y, T)
    emit ⟨R^a, D⟩
return E
```

by retaining a bootstrap sample of a percentage, B, of the instances in $\mathbf{W}'$. PCA is applied to this last matrix, $\mathbf{W}''$, to obtain a rotation matrix, $\mathbf{C}$.

In the reduction phase, all the resulting PCA rotation matrices, K, are arranged into a block diagonal matrix, $\mathbf{R}$, although it cannot yet be used to rotate the original input data, $\mathbf{X}$, because the order of its columns does not match the order of the corresponding features in the original data. In consequence, the columns of $\mathbf{R}$ have to be rearranged to match the original order of features using a permutation matrix, $\mathbf{P}$, and a MapReduce implementation of matrix multiplication.

Finally, a Random Forest classifier is trained using the data obtained by rotating the input data, $\mathbf{X}$, using the reorganized matrix, $\mathbf{R}^a$ ($\mathbf{X}\,\mathbf{R}^a$).

Figure 1 illustrates the process of calculating the rotation matrix $\mathbf{R}^a$ using as an example a data set $\mathbf{X}$ with 12 instances, 6 features, and 3 classes $\{0, 1, 2\}$; and algorithm parameters, K and B, equal to 3 and to 50%, respectively.

A thorough experimentation was performed taking into account both accuracy and execution time, using several representative data sets. The Rotation Forest classifier implementation for Spark is publicly available at https://github.com/mjuez/rotation-forest-spark.

Modern Bayesian tests were used for evaluating the statistical differences between the different algorithms that were tested. Our experiments demonstrated that Rotation Forest is a better option than Random Forest for massive data sets. Furthermore, its superiority in relation to PCARDE [2], a very recent ensemble algorithm for Big Data, has been demonstrated.

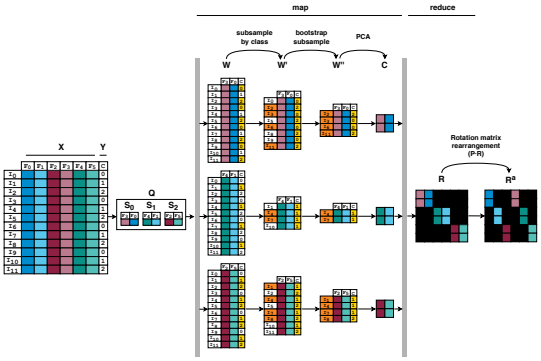


Fig. 1. MapReduce implementation of the process to generate the rotation matrix ($\mathbf{R}^a$).

By an experimental exploration of some algorithm parameters we concluded that the approach for training Rotation Forest with Random Forest rather than of a single decision tree as the base classifier proved that accurate models with fewer data rotations, and therefore models that train and predict faster, are indeed feasible. The analysis also reported that small ensembles, consisting of 10 trees, are accurate enough for the Big Data sets used in the study.

Additionally an evaluation of the influence of the bootstrap sample size with huge data sets has been conducted. The conclusion of that evaluation was that sampling 10% of the data provided classification models with an equivalent performance to those that sampled 25% or 50% of the data. The use of low percentages therefore means simpler PCA calculations and faster training of the Rotation Forest.

Rotation Forest code has been carefully developed following the Spark ML API guidelines, aiming towards its incorporation in the API within the near future. The number of ensemble algorithms available for Big Data is still scarce, specially for tasks such as online learning and unbalanced data sets, among others.

References

[1] Alessio Benavoli, Giorgio Corani, Janez Demšar, and Marco Zaffalon. Time for a change: a tutorial for comparing multiple classifiers through bayesian analysis. *The Journal of Machine Learning Research*, 18(1):2653–2688, 2017.

[2] Diego García-Gil, Sergio Ramírez-Gallego, Salvador García, and Francisco Herrera. Principal components analysis random discretization ensemble for big data. *Knowledge-Based Systems*, 150:166 – 174, 2018.

[3] Mario Juez-Gil, lvar Arnaiz-Gonzlez, Juan J. Rodrguez, Carlos Lpez-Nozal, and Csar Garca-Osorio. Rotation forest for big data. *Information Fusion*, 74:39–49, 2021.

[4] Ludmila I Kuncheva. *Combining pattern classifiers: methods and algorithms*. John Wiley & Sons, 2014.

[5] Juan José Rodriguez, Ludmila I Kuncheva, and Carlos J Alonso. Rotation forest: A new classifier ensemble method. *IEEE transactions on pattern analysis and machine intelligence*, 28(10):1619–1630, 2006.

Rotation Forest for Multi-target Regression*

Juan J. Rodríguez, Mario Juez-Gil, Carlos López-Nozal, Álvar Arnaiz-González
Dpto. Ingeniería Informática, Universidad de Burgos
Escuela Politécnica Superior, Avda. Cantabria s/n, 09006, Burgos, Spain

***Abstract*—This is a summary of our article published in Int. Journal of Machine Learning and Cybernetics [1], presented to the Multi-Conference CAEPIA'21 KeyWorks.**

The task of predicting multiple numeric outputs at the same time is called Multi-Target Regression (MTR). MTR poses additional difficulties to traditional single-target regression (STR), and many real-world problems involve the prediction of multiple targets at once. One of the most successful approaches to deal with MTR, among others, consists in transforming the problem in several STR problems, whose outputs will be combined building up the MTR output. In this paper, the Rotation Forest ensemble method, previously proposed for single-target regression, is adapted to MTR tasks and tested with several regressors and data sets.

***Index Terms*—multi-target regression, ensemble, rotation forest**

I. Summary

The prediction of multiple numeric outputs at the same time is called Multi-Target Regression (MTR), and it has gained attention during the last decades. One of the most successful approaches to deal with MTR, although not the only one, consists in transforming the problem in several Single-Target Regression (STR) problems, whose outputs will be combined building up the MTR output.

In this paper, the Rotation Forest [2] ensemble method, is adapted to MTR tasks and tested with several regressors and data sets. Our proposal rotates the input space in an efficient and novel fashion, avoiding extra rotations forced by MTR problem decomposition.

The keystone of Rotation Forest relies on transforming the input space by means of Principal Component Analysis (PCA), what makes it more powerful than other ensemble approaches such as Bagging or Random Forest. Rotation Forest for STR can be used for MTR using problem transformation methods. These methods convert original MTR problems into several STR problems. The drawback of using MTR transformation methods with Rotation Forest for STR is that as many rotations of the data sets are required as the number of STR tasks multiplied by the ensemble size. Nevertheless, the rotated data sets for one target are also valid for the other targets, implying some wasted effort.

We would like to thank *Ministerio de Economía y Competitividad* of the Spanish Government for financing the project TIN2015-67534-P (MINECO/-FEDER, UE), to the *Junta de Castilla y León* for financing the project BU085P17 (JCyL/FEDER, UE) (both projects co-financed through European Union FEDER funds), and to the *Consejería de Educación of the Junta de Castilla y León* and the European Social Fund for the EDU/1100/2017 predoctoral grant. The authors gratefully acknowledge the support of the NVIDIA Corporation and its donation of the TITAN Xp GPUs used in this research.

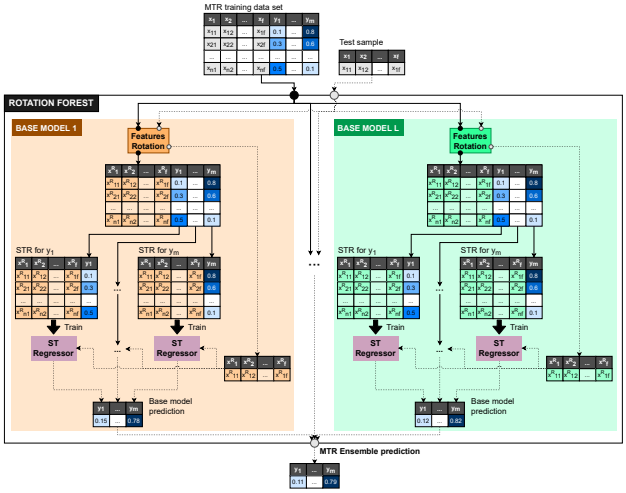


Fig. 1. An example of the Rotation Forest ensemble proposed in this paper, where data set rotation is performed before applying the MTR approach. All single-target tasks of a specific base model share the same rotation of the data (depicted using the same color).

The alternative approach considered in this paper is to use Rotation Forest as an ensemble method where each base model is an MTR model. In this way, the number of rotations of the data set will be limited to the size of the ensemble (e.g., 100), regardless of the number of targets and the approach used to deal with MTR (see Figure 1).

The Rotation Forest for MTR is presented in Algorithm 1. Each of the L models in the ensemble, D_t, is built with a different rotation of the training data with each rotation defined by a rotation matrix. The procedure for obtaining each rotation matrix, $\mathbf{R}_t^a$, is as follows. The features are divided into groups where the number of groups, K, is an argument of the method.

A data set, $\mathbf{X}_{t,k}$, is considered for each group, formed only of the features of each group, $\mathbf{F}_{t,k}$. In the Rotation Forest method for classification [2], this data set is filtered, removing all the instances of a proper subset of the classes. For MTR it is also desirable to select a subset of the instances according to the output values, a random projection of the outputs is taken and the selected instances are those with the lowest values in this projection.

A bootstrap sample, with a default sample size of 50% is taken from those selected instances. Then, Principal Component Analysis (PCA) is applied to the sample, generating a rotation matrix, $C_{t,k}$.

The removal of instances with the lowest values from a

TABLE I
AVERAGE RANKS FROM ALL THE CONSIDERED METHODS.

Method	Rank	Method	Rank	Method	Rank
RotF-ERC-train	9.6429	ERC-train-BagP	20.6786	SST-true-RanF	24.7143
RotF-SST-train	10.0000	RotRanF-ERC-train	20.8571	SST-cv-BagP	24.7500
RotF-ST	11.3571	ERC-cv-BagP	21.3214	SST-train-BagP	24.8214
RotF-ERC-cv	12.7500	RotRanF-SST-true	21.7857	ST-BagP	25.4643
RotF-SST-cv	12.9643	RotRanF-ERC-true	22.0357	RotRanF-eMRQ	26.3571
RotF-ERC-true	14.1071	RotRanF-SST-train	22.1071	ERC-true-BagP	26.9286
RotF-eMRQ	14.8214	RotRanF-ERC-cv	22.2143	SST-true-BagU	28.2857
RotF-SST-true	17.0357	SST-cv-RanF	22.2857	eMRQ-BagU	29.7500
ERC-train-RanF	17.5000	ST-RanF	22.5714	SST-true-BagP	29.8214
ERC-cv-RanF	18.3214	ERC-true-BagU	23.2500	eMRQ-RanF	31.9286
ERC-cv-BagU	18.3571	ST-BagU	23.8929	eMRQ-BagP	32.0714
ERC-train-BagU	19.3571	RotRanF-SST-cv	24.1429	RotRanF-MRQ	32.1429
ERC-true-RanF	19.6071	SST-train-BagU	24.2500	MRQ-BagU	39.6071
RotRanF-ST	19.8214	SST-cv-BagU	24.6786	MRQ-BagP	40.7500
SST-train-RanF	20.1071	RotF-MRQ	24.7143	MRQ-RanF	41.0714

Algorithm 1: Rotation Forest for MTR.

Input: A training set $(\mathbf{X}, \mathbf{Y})$ where $\mathbf{X} = \{\mathbf{x}_1, \dots \mathbf{x}_n\}$ and $\mathbf{Y} = \{\mathbf{y}_1, \dots \mathbf{y}_n\}$, ensemble size L, number of feature subsets K, base learner.
Output: Ensemble E

for $t \leftarrow 1, \dots, L$ **do**
  Prepare the rotation matrix $\mathbf{R}_t^a$:
  begin
    Randomly split $\mathbf{F}$ into K subsets $\mathbf{F}_{t,k}$
    for $k \leftarrow 1, \dots, K$ **do**
      $\mathbf{X}_{t,k} \leftarrow$ submatrix of $\mathbf{X}$ for the features in $\mathbf{F}_{t,k}$
      $\mathbf{X}'_{t,k} \leftarrow$ submatrix of $\mathbf{X}_{t,k}$ with the instances with smallest values in a random projection of $\mathbf{Y}$
      $\mathbf{X}''_{t,k} \leftarrow$ bootstrap sample from $\mathbf{X}'_{t,k}$
      $\mathbf{C}_{t,k} \leftarrow$ rotation matrix from $\mathsf{PCA}(\mathbf{X}''_{t,k})$
    $\mathbf{R}_t \leftarrow$ arrangement of the $\mathbf{C}_{t,k}$ matrices in a single rotation matrix
    $\mathbf{P}_t \leftarrow$ permutation matrix, matching the order of the features in $\mathbf{F}$
    $\mathbf{R}_t^a \leftarrow \mathbf{P}_t \mathbf{R}_t$ *// Rearrangement of* $\mathbf{R}_t$
  $D_t \leftarrow \mathsf{build\text{-}multi\text{-}target\text{-}regressor}(\mathbf{X}\mathbf{R}_t^a, \mathbf{Y})$
$E \leftarrow \bigcup_{t=1}^{L} D_t$ *// For* $\mathbf{x}$, E *predicts* $E(\mathbf{x}) = \frac{1}{L}\sum_{t=1}^{L} D_t(\mathbf{x})$

random projection and in the bootstrap sample is for the sake of diversity. The same subset of features can be selected for several ensemble models (especially if the number of features is low). If the rotations are obtained by means of PCA using all the training instances, then the rotated features should be the same.

We therefore have a rotation matrix for each group of features that defines a new set of features on the basis of the PCA components. All the components of all the groups are gathered together in a single rotation matrix, $\mathbf{R}_t$. This matrix could not be used directly on the training data, because the features are not in the same order, so the rotation matrix is rearranged using a permutation matrix, $\mathbf{P}_t$, that simply reorders the features, yielding the final rotation matrix, $\mathbf{R}_t^a$.

Finally, the training data set is rotated using the rotation matrix, then the rotated data set is used to build the base MTR model.

The objective of the experimentation is to validate the Rotation Forest for MTR proposal, and to compare its performance with other MTR ensembles. Regarding the different problem transformation approaches, other objective of this experimentation is to assess which performs better used along Rotation Forest.

28 data sets were used. Regarding methods, nine approaches for MTR were used: Single Target (ST); three variants of Stacked Single Target (SST-true, SST-train, and SST-cv); three variants of Ensembles of Regressor Chains (ERC-true, ERC-train, and ERC-cv); and Multi-target Regression via Quantization (MRQ) and the ensemble version of it (eMRQ).

The performance was measured using aRRMSE, the average Relative Root Mean Squared Error. The experiments were performed using Mulan [3]. The results were obtained using a 10-fold cross validation.

Table I shows the average rankings of the 45 methods tested. The eight top positions are for methods with RotF. The two top methods are RotF-ERC-train and RotF-SST-train, while the third method is RotF-ST.

When comparing the nine multi-target approaches using Rotation Forest, the best methods were RotF-SST-train and RotF-ERC-train. Fortunately, the most expensive methods with internal cross validation hardly appeared necessary.

In contrast, none of the methods showed a clear advantage over the most straightforward approach: RotF-ST. As it was noted, this method treats each output independently. Hence, it raises the question of whether the results of RotF-ST can be improved using other MTR approaches, considering output dependencies for building the models that are combined in Rotation Forest.

REFERENCES

[1] Juan J Rodríguez, Mario Juez-Gil, Carlos López-Nozal, and Álvar Arnaiz-González. Rotation forest for multi-target regression. *International Journal of Machine Learning and Cybernetics*, pages 1–26, 2021.

[2] Juan J. Rodríguez, Ludmila I. Kuncheva, and Carlos J. Alonso. Rotation forest: A new classifier ensemble method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(10):1619–1630, Oct 2006.

[3] Grigorios Tsoumakas, Eleftherios Spyromitros-Xioufis, Jozef Vilcek, and Ioannis Vlahavas. Mulan: A java library for multi-label learning. *Journal of Machine Learning Research*, 12:2411–2414, 2011.

Experimental evaluation of ensemble classifiers for imbalance in Big Data*

Mario Juez-Gil, Álvar Arnaiz-González, Juan J. Rodríguez, César García-Osorio
Dpto. Ingeniería Informática, Universidad de Burgos
Escuela Politécnica Superior, Avda. Cantabria s/n, 09006, Burgos, Spain

***Abstract*—This is a summary of our article published in Applied Soft Computing [3], presented to the Multi-Conference CAEPIA'21 KeyWorks.**

The size of datasets are growing in size and complexity at a pace never seen before, forming ever larger datasets known as Big Data. A common problem in Big Data classification is that datasets are usually imbalanced. Nowadays, although the number of imbalanced classification methods have increased, they continue to focus on normal-sized datasets and not on the new reality of Big Data. In this paper, in-depth experimentation with ensemble classifiers is conducted in the context of imbalanced Big Data classification, using two popular ensemble families (Bagging and Boosting) and different resampling methods.

***Index Terms*—Unbalance, Imbalance, Ensemble, Resampling, Big Data, Spark**

I. Summary

Datasets are growing in size and complexity at a pace never seen before, forming ever larger datasets known as Big Data. A common problem for classification, especially in Big Data, is that the numerous examples of the different classes might not be balanced. Some decades ago, imbalanced classification was therefore introduced, to correct the tendency of classifiers that show bias in favor of the majority class and that ignore the minority one. To date, although the number of imbalanced classification methods have increased, they continue to focus on normal-sized datasets and not on the new reality of Big Data.

In this paper, in-depth experimentation with ensemble classifiers is conducted in the context of imbalanced Big Data classification, using two popular ensemble families (Bagging and Boosting) and different resampling methods. All the experimentation was launched in Spark clusters, comparing ensemble performance and execution times with statistical test results, including the newest ones based on the Bayesian approach.

Simple sampling techniques, such as Random OverSampling (ROS) and Random UnderSampling (RUS) were among the first attempts to deal with imbalance in Big Data classification [1].

SMOTE is another popular algorithm to deal with imbalanced datasets, the straightforward implementation and sound performance of which has increased its popularity and led to a proliferation of SMOTE variants.

The ROSEFW-RF [5] algorithm (Random OverSampling and Evolutionary Feature Weighting for Random Forest) was the winning algorithm in the ECBDL'14 Big Data competition. This algorithm balances the classes using ROS and identifies the most relevant features through an evolutionary feature-selection process, before building a Random Forest classifier. The algorithm demonstrated its strengths winning the competition.

The lack of methods applicable to imbalanced learning in Big Data frameworks poses difficulties when extracting knowledge from large datasets with unevenly distributed classes. The scarcity of methods becomes even more noticeable when it is compared with the large number of methods for normal-sized datasets. Furthermore, some of these methods were implemented only for Hadoop before Spark became more popular, and thus, their comparison is even more challenging. To the best of our knowledge, no comprehensive experimentation has been conducted to date, to compare resampling-based ensemble methods with Big Data imbalance classification, and we consider that there is a need for that kind of experimental evaluation. Knowing whether resampling techniques can and to what extent they can benefit imbalanced Big Data classification is essential to accomplish meaningful and successful future research.

The experimental set-up was organized into two groups, in order to determine the effects of the balancing/resampling strategies on the performance of the ensembles:

- Dataset resampling and then training: the dataset was balanced once at the beginning (following one of the strategies) and an ensemble classifier was trained using the resampled dataset.
- Resampling within the ensemble: the ensemble performs a resampling strategy before the training of each base classifier (*i.e.*, there are as many resamples as there are base classifiers).

Five popular sampling techniques were tested in the experimentation: RUS, ROS, SMOTE, ROSE [4], and RB [2]. All the algorithms were implemented in Scala and executed within the Apache Spark framework.

The performance of the sampling techniques was compared using three different ensemble classifiers available for Spark: Bagging (BAG), Random Forest (RANF), and Gradient Boost-

We would like to thank the *Junta de Castilla y León* for financing the project BU055P20 (JCyL/FEDER, UE) cofinanced from European Union FEDER funds and the European Social Fund through a pre-doctoral grant (EDU/1100/2017). The research leading to these results has received funding from "La Caixa" Foundation under agreement LCF/PR/PR18/51130007. This material is based upon work supported by Google Cloud.

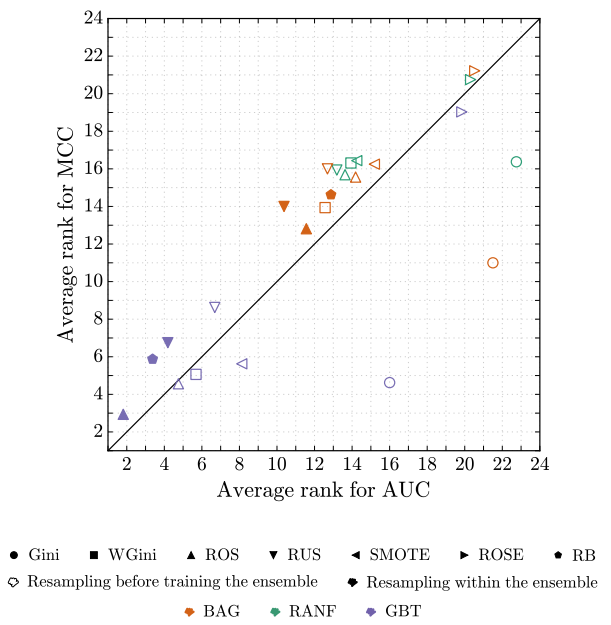


Fig. 1. Average ranks for all ensemble methods according to MCC and AUC metrics.

ing Trees (GBT), thus covering the Bagging and Boosting families of ensembles.

Figure 1 shows the performance of the methods by using average ranks with one-to-one comparisons: one metric on each axis. The marker aspect represents: the resampling strategy (shape), the ensemble method (color), and the two resampling strategies (unfilled means resampling before training, and filled means resampling within the ensemble). Two clusters could be found, one contained most of the GBT variants on the best positions (left lower corner), while the other contained RANF and BAG variants, which were located farther to the right. Differences between RANF and BAG were not so clear, but in general the BAG variants tended to perform better than the RANF ones. The differences between the metrics were remarkable for some methods, especially for the Gini variants (represented with circular markers), which were positioned far away from the diagonal line. This finding offers an interesting insight, insofar as the use of resampling may be not beneficial for the performance of some classifiers according to some metrics. For this reason, the use of several metrics is advisable and may even be crucial when drawing proper conclusions on imbalance within Big Data environments.

The rankings suggested to us that resampling before training had a fairly similar performance to resampling for each base classifier within the ensemble. The overall idea, was that no clear winner could be named. Neither could a similar performance between strategies be noted. Therefore, depending on the specific dataset one approach will be better than the other and vice-versa.

Within the experimental framework, Boosting ensembles, although requiring more computational power, clearly outperformed Bagging-based alternatives. The use on the trees construction of an impurity index that takes into account the imbalance, such as weighted Gini, offered roughly equivalent results to the use of resampling techniques. Surprisingly, the training of the ensembles on the original datasets without any change (using the standard Gini index), offered quite good results overall (for MCC and F_1-score metrics). However, this procedure is not advisable, because the results were dreadful for some datasets (clearly visible when AUC or G-mean were used), but still an accurate indicator of a lack of robust solutions for improving the performance for all the metrics. Regarding the resampling methods, ROS generally achieved better results, but with only a minimal advantage, closely followed by RUS and SMOTE with no statistically significant differences. In contrast, ROSE was clearly the worst alternative. Our conclusion is therefore that complex methods that involve the generation of synthetic instances are not as effective for Big Data as they are for normal-sized datasets. Although they could, depending on the dataset, be the best option, in general we discourage their use in favor of simpler and faster methods such as ROS.

Ensembles specifically designed to overcome the imbalance problem (*i.e.*, resampling before training each base classifier), achieved better performance than resampling a dataset once and then training a conventional ensemble with it. Nevertheless, the differences between the two strategies were very small, suggesting that whether one strategy is actually better than the other will strongly depend on the dataset to which it is applied. Therefore, whenever execution times are considered critical, as it is often the case with Big Data, the general recommendation would be to use the faster strategies based on a single initial resampling or the use of impurity indexes that take into account imbalance (*e.g.*, weighted Gini).

One very interesting conclusion from the study was that simpler methods applied to unbalanced datasets in the context of Big Data provided better results than complex methods. The additional complexity of some of the sophisticated methods, which appear necessary to process and to reduce imbalance in normal-sized datasets were not effective for imbalanced Big Data.

References

[1] Sara del Río, Victoria López, José Manuel Benítez, and Francisco Herrera. On the use of mapreduce for imbalanced big data using random forest. *Information Sciences*, 285:112 – 137, 2014.

[2] José F. Díez-Pastor, Juan J. Rodríguez, César García-Osorio, and Ludmila I. Kuncheva. Random balance: Ensembles of variable priors classifiers for imbalanced data. *Knowledge-Based Systems*, 85:96 – 111, 2015.

[3] Mario Juez-Gil, Álvar Arnaiz-González, Juan J. Rodríguez, and César García-Osorio. Experimental evaluation of ensemble classifiers for imbalance in big data. *Applied Soft Computing*, page 107447, 2021.

[4] Giovanna Menardi and Nicola Torelli. Training and assessing classification rules with imbalanced data. *Data Mining and Knowledge Discovery*, 28(1):92–122, 2014.

[5] Isaac Triguero, Sara del Ro, Victoria Lpez, Jaume Bacardit, Jos M. Bentez, and Francisco Herrera. ROSEFW-RF: The winner algorithm for the ECBDL'14 big data competition: An extremely imbalanced big data bioinformatics problem. *Knowledge-Based Systems*, 87:69 – 79, 2015. Computational Intelligence Applications for Data Science.

Talleres, Competiciones y Doctoral Consortium



XIX Conferencia de la Asociación Española para la Inteligencia Artificial

II Taller de Grupos de investigación españoles de IA en Biomedicina (IABiomed)



Methods and measures to quantify ICU patient heterogeneity

David Cuadrado
Universitat Rovira i Virgili
Intensive Care Unit, University Hospital Joan XXIII
Tarragona, Spain
david.cuadrado@estudiants.urv.cat

David Riaño
Universitat Rovira i Virgili
Tarragona, Spain
david.riano@urv.cat

Josep Gómez
Intensive Care Unit, University Hospital Joan XXIII
Tarragona, Spain

Alejandro Rodríguez
Intensive Care Unit, University Hospital Joan XXIII
Tarragona, Spain

María Bodí
Intensive Care Unit, University Hospital Joan XXIII
Tarragona, Spain

***Abstract*—Case mix complexity of ICU patients makes the daily prediction of their days to discharge (DTD) a complex task. The use of tools for heterogeneity analysis of the patients is necessary in order to make more precise DTD predictors. In the absence of documentation, we propose four methods, tested with patients admitted over four years in Hospital Joan XXIII, and their corresponding measures to quantify the heterogeneity of intensive patients. The results can provide a basis in the construction of DTD prediction models. This paper is a summary of the work published in [1] for academic purposes.**

Keywords—patient heterogeneity, case-mix, days to discharge prediction, intensive care unit, patient similarity

I. Introduction

Patients admitted in Hospital Intensive Care Units (ICU) comprise a wide case-mix and care variability [2] and require constant care and supervision. In order to have a comprehensive and comparative notion of the ICU patient an identification and quantification study of the variety of patients would be relevant. Throughout the stay, multiple patient data are recorded during the stay or at the beginning of admission. With the data obtained during the first 24-48 hours a Length of Stay (LOS) can be predicted in order to obtain the number of days to stay of a patient, while the Days To Discharge (DTD) prediction dynamically measure the number of days until the patient leaves the ICU and is predicted daily.

The two parameters have an impact on the clinical, administrative and management aspects and provide additional values to clinicians to make corrections on patients in order to modify actions. While LOS brings many benefits, DTD would also allow provide an improvement in planning, bed occupancy [3], complex surgeries requiring ICU [4], or transfers to other centers [5] depending on the result of the prediction. However, DTD prediction in the ICU has received little attention in the literature [6], possibly due to the difficulty of collecting daily data or to the sophistication of the technologies for longitudinal data analysis.

An initial understanding of the heterogeneity of ICU patients would allow an improvement in the identification of patients for a future prediction of DTD. We have developed four methods with their corresponding measures to quantify the heterogeneity of ICU patients.

The first method (clinical parameter analysis) assumes that as we get closer to the day of discharge, the clinical values tend to "normalize" until they reach a characteristic state of the discharged patient while their variability decreases. The second method (severity scales analysis) is a variation of the first. In this case we assume that some clinical scales measuring patient's condition, such as SOFA-Total, NAS, or EMINA, tend to "normalize" until reaching a characteristic state of the patient. The third method (confusion analysis) use a similarity function to calculate the difference between patients discharged in x days to patients discharged in y days. The results expressed in a confusion matrix show the percentage of patients similar to those discharged before or after i days. The fourth method (cluster analysis) uses internal evaluation methods to evaluate quality of DTD groups. The three methods used for the evaluation are Davies-Bouldin [7], Dunn [8], and silhouette [9].

All methods have been applied in patients discharged alive and admitted between 2016 and 2019, prior to COVID-19. The daily information of these patients in the 21 days previous to discharge was used in the analysis.

II. METHODS

Four methods to measure ICU patients heterogeneity were formalized, with the usage of Python for its implementation and test.

For the analysis of heterogeneity we have a database composed with N ICU patients P_1, ..., P_N discharged alive where every patient P_i (i = 1, ..., N) contains a sequence of daily descriptions of the patient in terms of m clinical parameters, which can be numerical or categorical. All clinical parameters are normalized, using its maximum and minimum values, in an interval [0,1]. Every daily description is defined by a DTD group DTD_j where j corresponds to the j-th day before discharge. Similarity between two normalized values v and v‘ for numerical parameter is $sim_j(v, v') = 1-(v-v')^2$ and similarity between two values v and v‘ for categorical parameter is $sim_j(v, v') = 1$, if $v = v'$, or 0 if $v \neq v'$.

A. Method 1: clinical parameters analysis

As patient's day of discharge approaches, clinical parameters should converge to normality values and their variability should decrease. For every DTD group, heterogeneity is measured by the deviation from average values from the normality values or range and how much of the fluctuation of these values decrease as approaches to discharge day.

B. *Method 2: severity scales analysis*

Severity scales simplify medical interpretation of patient's organ failure (SOFA-Total), risk of developing ulcers (EMINA) or percentage of nursing activity (NAS) measuring the complexity of ICU patients, condition severity, or care requirements. A graphical representation of means values and standard deviation for every severity scale grouped by DTD should provide values converging to normality values and variability decreasing as day to discharge approaches.

C. *Method 3: confusion analysis*

A patient similarity function combined with a threshold parameter $\delta \in [0,1]$ can be used to identify similarity between patients from DTD_i and DTD_j (i, j = 1, ..., l). Results are reflected in a $n_\delta(i,j)$ matrix with the proportion of cases in DTD_i which can be confused with patients in DTDj. Heterogeneity is measured by the confusing discharges reflected in the matrix. Four confusion ratios measure the heterogeneity as a degree of confusion, premature discharge, overdue discharge, and feasible discharge error, respectively. Finally, we calculate the average number of patients from DTD_i which are similar to patients from DTDj where $i \neq j$.

D. *Method 4: cluster analysis*

Three methods for interpreting and validate clusters (based on premise that elements from same cluster are similar and elements from different clusters are dissimilar) are used to convert distance-based indices into ICU patient similarity indices. These three methods, Davies-Bouldin, Dunn and silhouette, provide information on the degree of heterogeneity of the patients in the DTD groups based on their indices.

TABLE I. COMPARISION OF METHODS AND CONCLUSIONS

Methods	Conclusions
Method 1	SOFA-Cardio, SOFA-CNS and SOFA-Resp averages decrease as discharge day aproaches but other parameters such as MAP or Tmp shows increases at discharge or remains with high variability
Method 2	SOFA-Total, NAS, and EMINA averages decrease as discharge day aproaches but variability rest above 10%
Method 3	A large proportion of patients from a DTD group are similar to other patients from different DTD group
Method 4	Results conclude that there are difficulties to differentiate among patients with different DTD

III. RESULTS AND DISCUSSION

The evolution of the mean clinical parameters of ICU patients represented in a graph allows us to observe the evolution of the parameters from DTD21 to DTD1. While some of the parameters such as SOFA-Cardio, SOFA-CNS, and SOFA-Resp show a decrease as the departure day approaches, other values such as mean arterial pressure (MAP) increase. Parameters such as SOFA-Cardio and SOFA-CNS, and SOFA-Resp show a variability above 25% until the day before discharge while the rest of parameters show a variability in the last 10 days between 10% and 25%, with the exception of the temperature whose variability is always above those parameters.

Patient heterogeneity based on severity scales shows reduction of mean values of SOFA-Total, NAS, and EMINA in last days before discharge while variability decrease but remains above 10%.

A confusion analysis based on a $n_\delta(i,j)$ matrix and the confusion measures for $\delta = 0.9$ (meaning that two patient descriptions are similar only if a similarity of 0.9 or higher is reached) show the largest values in the diagonal and a decrease as we move away the diagonal. This means that there is a degree of confusion between patients that are discharged in close days, and this confusion decreases as the difference between days to discharge gets larger. Results analyzed confirm 37% (on average) patients closely resemble the patients discharged earlier, and 26% (on average) patients are very similar to patients who were discharged later. 63% of the patients discharged one day resemble patients that where discharged other days, on average.

Davies-Bouldin, Dunn, and silhouette indices for DTD groups obtain the values 11.43, 0.037, and -0.054. An interpretation of these results [10] concludes that patients with similar DTD are not necessarily similar and patients with different DTD values are not necessarily different.

The inclusion of all the clinical cases seen in the ICU including all the survival cases in four consecutive years makes the study possibly representative of many other ICUs. Multiple types of ICU patients are involved. Restricting the study to a reduced type of patients (e.g., surgical or scheduled) could affect the results on heterogeneity. Finally, some patient discharges may have been based on organizational reasons rather than pure clinical reasons, affecting the final results.

IV. CONCLUSIONS

Table 1 summarizes the conclusions. The four methods applied confirm high heterogeneity in patients with same DTD group and a confusion between patients with different DTD. This may confirm the high complexity of making accurate DTD predictors.

[1] D. Cuadrado, D. Riaño, J. Gómez, M. Bodí, A. Rodríguez, Methods and measures to quantify ICU patient heterogeneity, Journal of Biomedical Informatics 117 (2021) 103768

[2] M. C. Lenert, R. A. Miller, Y. Vorobeychik, C. G. Walsh, A method for analyzing inpatient care variability through physicians orders, Journal of Biomedical Informatics 91 (2019) 103111.

[3] J. Ruyssinck, J. van der Herten, R. Houthooft, F. Ongenae, I. Couckuyt, B. Gadeyne, K. Colpaert, J. Decruyenaere, F. D. Turck, T. Dhaene, Random survival forests for predicting the bed occupancy in the intensive care unit, Computational and Mathematical Methods in Medicine (2016) 1–7.

[4] Y. U. Bing-Hua, Delayed admission to intensive care unit for critically surgical patients is associated with increased mortality, The American Journal of Surgery 208 (2) (2014) 268–274.

[5] J. M. Droogh, M. Smit, A. R. Absalom, J. J. Ligtenberg, J. G. Zijlstra, Transferring the critically ill patient: are we there yet?, Critical Care 19 (1) (2015) 62.

[6] D. Cuadrado, D. Riaño, J. Gómez, M. Bodí, G. Sirgo, F. Esteban, R. Rodríguez, Pursuing optimal prediction of discharge time in ICUs with machine learning methods, in: Artificial Intelligence in Medicine, Vol. 11526, Springer, 2019, pp. 150–154.

[7] D. L. Davies, D. W. Bouldin, A cluster separation measure, IEEE Transactions on Pattern Analysis and Machine Intelligence 1 (2) (1979) 224–227.

[8] J. C. Dunn, Well-separated clusters and optimal fuzzy partitions, Journal of Cybernetics 4 (1) (1974) 95–104.

[9] P. J. Rousseeuw, Silhouettes: A graphical aid to the interpretation and validation of cluster analysis, Journal of Computational and Applied Mathematics 20 (1987) 53–65.

[10] M.Halkidi, Y.Batistakis, M.Vazirgiannis, On clustering validation techniques, Journal of Intelligent Information Systems 17(2001)107–14.

Neural Style Transfer to deal with the Domain Shift Problem on Glioblastoma Segmentation

Manuel García-Domínguez, César Domínguez, Jónathan Heras, Eloy Mata, and Vico Pascual
dept. of Mathematics and Computer Science
University of La Rioja
Logroño, Spain
{manuel.garciad, cesar.dominguez, jonathan.heras, eloy.mata, vico.pascual}@unirioja.es

***Abstract*—Domain shift is a generalisation problem of machine learning models that occurs when the data distribution of the training set is different to the data distribution encountered by the model when it is deployed. This is common in the context of biomedical image segmentation due to the variance of experimental conditions, equipment, and capturing settings. In this work, we address this challenge by studying neural style transfer algorithms in the context of the segmentation of tumour spheroids. We show that, using these algorithms, it is possible to recover the performance of a model that suffers from the domain shift problem. Finally, we provide an API to facilitate the use neural style transfer techniques in the bioimaging context.**

***Index Terms*—Domain adaption, Style Transfer, Semantic Segmentation, Glioblastoma, Spheroids**

I. Introduction

Deep convolutional neural networks have become the state-of-the-art approach to tackle segmentation problems in medicine [19], [22]. However, there are several challenges that hinder the training and deployment of deep learning models in this context. First of all, a considerable amount of annotated images is needed to train a deep model, and annotating datasets for image segmentation is a tedious and time-consuming task that requires expert knowledge [13]. Moreover, there is an important generalisation challenge when using trained models that is known as *domain shift* (also known as distribution shift) [2], [4]. This problem arises when the data distribution of the training dataset employed for training a model is different to the data that the model encounters when deployed. This is common in biomedical datasets since images greatly vary due to experimental conditions, and the equipment (for instance, microscopes) and settings (for instance, focus and magnification) employed for capturing those images.

This generalisation problem can be tackled by combining datasets from multiple sources [7] or using techniques like data augmentation [21]; nevertheless, it is not possible to foresee every new and unknown distribution. A different approach consists in applying transfer learning [18], a technique that, instead of training a model from scratch, reuses a model pre-trained in a source dataset to train a new model in a target dataset. However, this requires the annotation of the target dataset, a time-consuming task that should be carried out for every new dataset. A different approach to handle the domain shift problem is image-to-image translation [8], a set of techniques that aim to learn the mapping between an input image and an output image using a training set of aligned image pairs; however, this requires paired data from the source and target domains, a challenge that can be faced by using unpaired image-to-image translation [26].

Unpaired image-to-image translation methods translate an image from a domain A to a domain B, and vice versa, in the absence of paired examples. This approach has been already employed in several medical segmentation tasks; for instance, the segmentation of the left ventricle in magnetic resonance images [24], the segmentation of digitally reconstructed radiographs [25], and the segmentation of magnetic resonance imaging (MRI), abdominal CT and MRI, and mammography X-rays [9]. All these works are based on variants of CycleGAN [26], an unpaired image-to-image translation method based on Generative Adversarial Networks (GANs) that requires two datasets: one of them contains images from the distribution employed for training the segmentation model, and the other contains images acquired in a different setting. This approach poses two challenges. First, both datasets must be available, and this might be an issue due to privacy concerns [1]; and, secondly, CycleGAN variants must be trained, a process that demands the usage of GPUs and might be challenging for several users due to the difficulties of training GAN models [20]. The approach proposed in this paper to tackle these drawbacks consists in using style transfer methods [5]; that is, techniques that render the content of an image using the style of another. Those techniques do not require a training process, and it is enough with releasing one image of the dataset employed for training the model that suffers the domain shift problem.

In this work, we have 3 studied neural style transfer techniques to deal with the domain shift problem in the context of segmenting tumour spheroids [15]. In this task, we have observed, see Section II, that models that achieve a mean IoU over 97% when evaluating with data following the same distribution as the training set, fail when they are employed with data following a different distribution (the IoU is, in some cases, under 15%). We have faced this domain shift problem by using style transfer techniques. Namely, the contributions of our work are:

This work was partially supported by Ministerio de Economía y Competitividad [MTM2017-88804-P], and a FPI Grant of Community of La Rioja

Dataset	♯ Images	Image size	Microscope	Magnification	Format	Type
BL5S	50	1296× 966	Leica	5x	TIFF	RGB
BN2S	154	1002× 1004	Nikon	2x	ND2	Gray 16bits
BN10S	105	1002× 1004	Nikon	10x	ND2	Gray 16bits
BO10S	64	3136 × 2152	Olympus	10x	JPG	RGB

TABLE I
FEATURES OF THE 4 DATASETS EMPLOYED IN THIS WORK. THE FORMER THREE DATASETS WERE EMPLOYED FOR TRAINING AND THE LAST DATASET WAS USED FOR TESTING

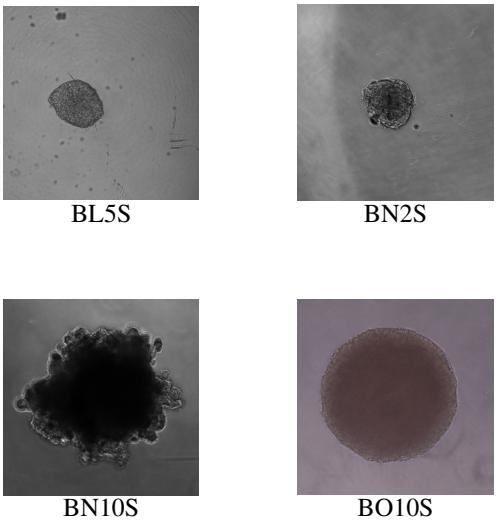


Fig. 1. Samples from the 4 datasets employed in this work

- We explore several state-of-the-art style transfer translation methods to tackle the domain shift problem in the context of tumour spheroid segmentation.
- We demonstrate the effectiveness of using style transfer to improve the performance of a variety of advanced deep segmentation networks.
- We provide an API to apply the studied methods not only in the context of spheroid segmentation but in general for medical imaging tasks. The API is available at https://github.com/ManuGar/ImageStyleTransfer

II. MATERIALS

Spheroids are the most widely used 3D models to study cancer since they can be used for studying the effects of different micro-environmental characteristics on tumour behaviour and for testing different preclinical and clinical treatments [15]. The images from tumour spheroids greatly vary depending on the experimental conditions, and also on the equipment (microscopes) and conditions (focus and magnification) employed to capture the images [11].

For our experiments, we have employed the 4 datasets presented in [11]; a description of those datasets is provided in Table I, and an image of each dataset is shown in Figure 1. As can be noticed from Table I and Figure 1, there are considerable differences among the images of each dataset. Three of those datasets (the BL5S, BN2S, and BN10S datasets) were employed for training 4 segmentation models (using the algorithms DeepLab v3 [3], HRNet Seg [23], U-net [19] and U^2-Net [17]) and the last dataset (the BO10S dataset) was employed for testing. We have used this dataset split because the last dataset comes from a different laboratory; so, its style will not be the same as the others. The definition of the 4 segmentation architectures is available in the SemTorch package[1]. All the architectures were trained with the libraries PyTorch [16] and FastAI [6] and using a GPU Nvidia RTX 2080 Ti. In order to set the learning rate for the different architectures, we employed the procedure presented in [6]; and, we applied early stopping when training all the architectures to avoid overfitting. The metric employed to measure the accuracy of the different methods is the IoU, also known as Jaccard index — this metric measures the area of intersection between the ground truth and the predicted region over the area of union between the ground truth and the predicted region.

When the models were evaluated using a test set formed by images following the same distribution than the training set, the 4 models achieved a performance over 97%, see Table II. On the contrary, when those models were employed with images captured under different conditions (namely, using the BO10S dataset), the performance of the models decreased by up to 84%. In the next section, we explore how style transfer methods can serve to deal with the domain shift problem in this context.

[1]The SemTorch package is available at https://github.com/WaterKnight1998/SemTorch

	DeepLab v3	HRNet-Seg	U-Net	U2-Net
BL5S-BN2S-BN10S	97.00	97.32	97.25	97.26
BO10S	83.61	92.65	13.64	95.65

TABLE II
PERFORMANCE OF THE 4 MODELS WHEN EVALUATING IN A TEST SET FORMED FROM IMAGES FOLLOWING THE SAME DISTRIBUTION THAN THE TRAINING SET (BL5S-BN2S-BN10S), AND WHEN EVALUATED USING A TEST SET FROM A DIFFERENT DISTRIBUTION (BO10S)

III. STYLE TRANSFER

This section is devoted to present how style transfer methods can handle the domain shift problem. In addition, we introduce the API that we have developed to facilitate the use of those methods.

We start by explaining the procedure to apply style transfer methods to deal with the domain shift problem of a model — such a procedure is summarised in Figure 2. We assume that a model has been trained using a source dataset of images, and we are interested in applying such a model to obtain the prediction associated with an image from a different distribution than the source dataset; we call this image, the target image. Instead of feeding the target image directly to the model, we first take an image from the source dataset and transfer the style of that image to the target image but preserving its content producing a transformed image. Finally, the transformed image is fed to the model to obtain the associated prediction.

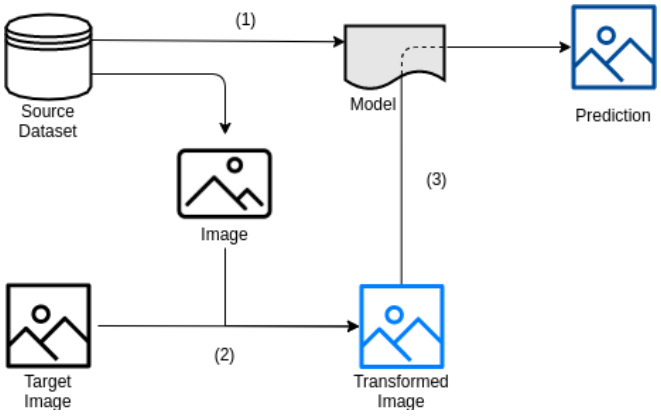


Fig. 2. Workflow of the style transfer approach. (1) A model is trained using a source dataset. (2) The target image is transformed using the style from an image of the source dataset. (3) The transformed image is fed to the model.

	DeepLab v3	HRNet-Seg	U-Net	U^2-Net
Base	83.61	92.65	13.64	95.65
NST	95.64↑	94.91↑	89.21↑	95.89↑
Deep Image Analogy	0.00↓	45.13 ↓	0.66↓	0.84↓
STROTSS	94.86↑	92.38↓	78.08↑	94.14↓

TABLE III
PERFORMANCE FOR THE BO10S DATASET USING THE DIFFERENT STYLE-TRANSFER METHODS TO DEAL WITH THE DOMAIN SHIFT PROBLEM. A ↑ INDICATES AN IMPROVEMENT WITH RESPECT TO THE BASE MODEL, WHEREAS A ↓ INDICATES A DECLINATION IN THE PERFORMANCE.

The key component of the aforementioned process is the algorithm that transfers the style from the source dataset but keeping the content of the target image. In the literature, there are several style transfer algorithms [14]; but, for our experiments, we have focused on three of them: neural style transfer (NST) [5], an optimisation technique that uses a Convolutional Neural Network (CNN) to decompose the content and style from images; deep image analogy [12], a method that finds semantically-meaningful correspondences between two input images by adapting the notion of image analogy with features extracted from a CNN; and STROTSS [10], a variant of the NST algorithm that changes the optimisation objective of NST.

It is worth noting that the style transfer approach presented here can be applied to deal with the domain shift problem not only for segmentation problems, as in our work, but also to other computer vision tasks. Hence, these methods can be helpful for a great variety of problems. However, it might be difficult to apply these techniques since they are implemented in different libraries and using different frameworks, and each of them has its own particularities. In this work, we have addressed this drawback by developing a high-level Python API that allows the integration of style transfer algorithms independently of their underlying library and framework. The API currently includes the aforementioned methods (the project webpage provides information about the library that implements each method) and can be easily extended with new techniques. In order to apply the previously introduced procedure using our API, users only have to provide the style image, the target image, and the name of the algorithm to apply; the rest of the transformation process is automatically conducted by the API.

IV. Results and discussion

In our running example of segmenting tumour spheroids, and using our API, we randomly picked an image from the combination of the datasets BL5S, BN2S, and BN10S, and used it to transform the images from the BO10S dataset. Subsequently, we fed those images to the segmentation models presented in Section II, and evaluated their performance, see Table III. From the 3 studied style transfer algorithms, both the NST and STROTSS algorithms handle the domain shift problem; whereas, the images transformed with the deep image analogy algorithm produce even worse results than the original images from the BO10S dataset. Using the NST algorithm, all the segmentation models improve their IoU (the U-Net model improves its performance from 13.64% to 89.21%, and the other models have an IoU close to 95%). For the STROTSS algorithm, the results are also positive: two of the segmentation models improve (DeepLab and U-Net), and the other two achieve worse results, but still their IoU is over 92%.

We can also visually inspect the images produced by the different transformation algorithms to discover the difficulties faced by the segmentation models, see Figure 3. We can notice that the 2 successful models (NST and STROTSS) produce images that preserve the content of the image but with a style that is similar to the style of those used for training the segmentation models. On the contrary, the deep image analogy method does not keep the content of the image; and, thus the segmentation models are not able to properly segment the images.

From Figure 3, we can also appreciate the sensibility of the segmentation models to variations in the input image. The HRNet Seg and U^2-net models are more robust than the DeepLab and U-net models — recall that all the models achieved an IoU over 97% when evaluating in data from the distribution of the training set. Hence, the style transfer methods can be employed not only to deal with the domain shift problem of computer vision models, but also to evaluate the robustness of such models.

V. Conclusions

In this paper, we have studied the benefits of applying style transfer techniques to deal with the domain shift problem in the context of tumour segmentation. The results show us that, using those translation methods, it is possible to recover the performance of a model that suffers from the domain shift problem. In contrast with other existing methods that deal with the domain shift problem, such as image-to-image translation models, style transfer algorithms have the advantage of not requiring a training step, and can be deployed by providing a single image from the source dataset.

In this work, we have only used style transfer techniques, but it remains as future work to compare these methods

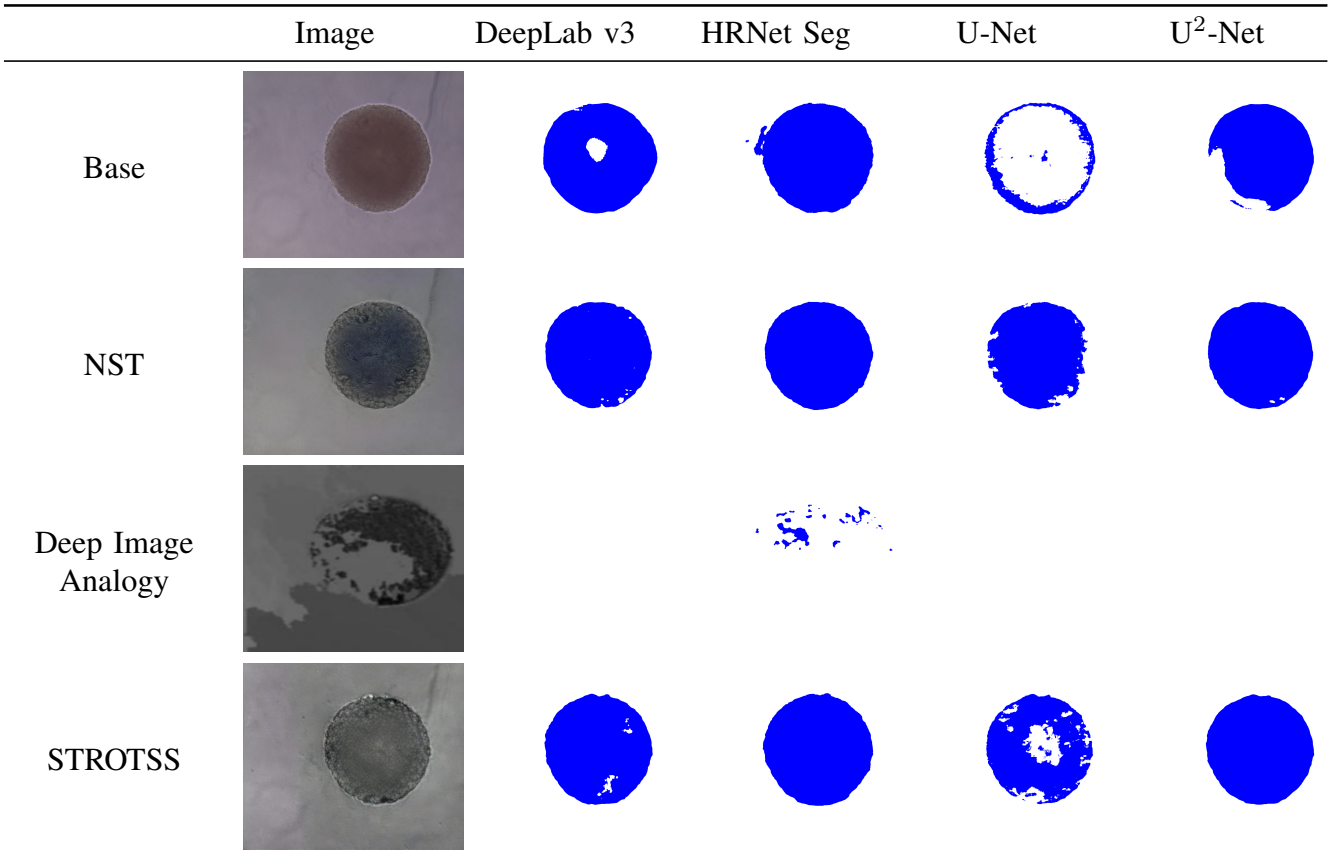


Fig. 3. An example showing the segmentation produced by the DeepLab, HRNet, U-Net and U^2-Net models after applying a style transfer algorithm or an image-to-image translation model to a given image

with unpaired image-to-image translation methods and other techniques that deal with domain shift. Since, it is known that not all algorithms work equally well for all problems, we plan to extend our API to include several image-to-image methods using GANs. Finally, we will test the performance of both neural style transfer methods and unpaired image-to-image models with other kinds of images and computer vision tasks.

References

[1] Adler-Milstein, J., Jha, A.K.: Sharing clinical data electronically: a critical challenge for fixing the health care system. JAMA **307**(16), 1695–1696 (2012). https://doi.org/10.1001/jama.2012.525

[2] Arvidsson, I., et al.: Generalization of prostate cancer classification for multiple sites using deep learning. In: IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018). pp. 191–194. IEEE (2018). https://doi.org/10.1109/ISBI.2018.8363552

[3] Chen, L.C., Zhu, Y., Papandreou, G., et al.: Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Computer Vision – ECCV 2018. pp. 833–851. Springer International Publishing (2018). https://doi.org/10.1007/978-3-030-01234-2_49

[4] Choudary, A., et al.: Advancing medical imaging informatics by deep learning-based domain adaptation. Year book of medical informatics **29**(1), 129–138 (2020). https://doi.org/10.1055/s-0040-1702009

[5] Gatys, L., Ecker, A., Bethge, M.: A neural algorithm of artistic style. Journal of Vision **16**, 326 (2016). https://doi.org/10.1167/16.12.326

[6] Howard, J., Gugger, S.: Deep Learning for Coders with FastAI and Pytorch. O'Reilly (2020)

[7] Irvin, J., Rajpurkar, P., Ko, M., et al.: Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In: Thirty-Third AAAI Conference on Artificial Intelligence. AAAI'19, vol. 33, pp. 590–597 (2019)

[8] Isola, P., et al.: Image-to-image translation with conditional adversarial networks. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5967–5976. IEEE (2017). https://doi.org/10.1109/CVPR.2017.632

[9] Jinzheng, C., et al.: Towards cross-modal organ translation and segmentation: A cycle- and shape-consistent generative adversarial network. Medical Image Analysis **52**, 174 – 184 (2019). https://doi.org/10.1016/j.media.2018.12.002

[10] Kolkin, N., Salavon, J., Shakhnarovich, G.: Style transfer by relaxed optimal transport and self-similarity. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 10043–10052. IEEE (2019). https://doi.org/10.1109/CVPR.2019.01029

[11] Lacalle, D., et al.: SpheroidJ: An Open-Source Set of Tools for Spheroid Segmentation. Computer Methods and Programs in Biomedicine (2021). https://doi.org/10.1016/j.cmpb.2020.105837

[12] Liao, J., et al.: Visual attribute transfer through deep image analogy. ACM Transactions on Graphics **36**(4), 120:1–120:15 (2017). https://doi.org/10.1145/3072959.3073683

[13] Litjens, G., et al.: A survey on deep learning in medical image analysis. Medical Image Analysis **42**, 60–88 (2017). https://doi.org/10.1016/j.media.2017.07.005

[14] Liu, L., et al.: Advanced deep learning techniques for image style transfer: A survey. Signal Processing: Image Communication **78**, 465–470 (2019). https://doi.org/10.1016/j.image.2019.08.006

[15] Nath, S., Devi, G.R.: Three-dimensional culture systems in cancer research: Focus on tumor spheroid model. Pharmacology & Therapeutics **163**, 94–108 (2016). https://doi.org/10.1016/j.pharmthera.2016.03.013
[16] Paszke, A., Gross, S., Massa, F., et al.: Pytorch: An imperative style, high-performance deep learning library. In: Advances in Neural Information Processing Systems 32, pp. 8024–8035. Curran Associates, Inc. (2019)
[17] Qin, X., Zhang, Z., Huang, C., et al.: U^2-net: Going deeper with nested u-structure for salient object detection. Pattern Recognition **106**, 107404 (2020). https://doi.org/10.1016/j.patcog.2020.107404
[18] Razavian, A.S., Azizpour, H., Sullivan, J., et al.: CNN features off-the-shelf: An astounding baseline for recognition. In: IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 512–519. CVPRW'14, IEEE (2014)
[19] Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. Lecture Notes in Computer Science, vol. 9351, pp. 234–241 (2015). https://doi.org/10.1007/978-3-319-24574-4_28
[20] Salimans, T., et al.: Improved techniques for training gans. In: 30th International Conference on Neural Information Processing Systems. pp. 2234–2242. NIPS'16, Curran Associates Inc. (2016)
[21] Simard, P., Steinkraus, D., Platt, J.C.: Best practices for convolutional neural networks applied to visual document analysis. In: Proceedings of the 12th International Conference on Document Analysis and Recognition (ICDAR'03). vol. 2, pp. 958–964 (2003)
[22] Sirinukunwattana, K., et al.: Gland segmentation in colon histology images: the glas challenge contest. Medical Image Analysis **35**, 489–502 (2017). https://doi.org/10.1016/j.media.2016.08.008
[23] Wang, J., Sun, K., Cheng, T., et al.: Deep high-resolution representation learning for visual recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence (2020). https://doi.org/10.1109/tpami.2020.2983686
[24] Yan, W., et al.: The domain shift problem of medical image segmentation and vendor-adaptation by unet-gan. In: Medical Image Computing and Computer Assisted Intervention (MICCAI 2019). pp. 623–631. Springer International Publishing (2019). https://doi.org/10.1007/978-3-030-32245-8_69
[25] Zhang, Y., et al.: Task driven generative modeling for unsupervised domain adaptation: Application to x-ray image segmentation. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 599–607 (2018)
[26] Zhu, J.Y., et al.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: IEEE International Conference on Computer Vision (ICCV 2017). pp. 2242–2251. IEEE (2017). https://doi.org/10.1109/ICCV.2017.244

Fall detection for healthy and autonomous elderly people*

Mirko Fáñez
Computer Science Department
University of Oviedo
Oviedo, Spain
mirko@mirkoo.es

José R. Villar
Computer Science Department
University of Oviedo
Oviedo, Spain
villarjose@uniovi.es

Enrique de la Cal
Computer Science Department
University of Oviedo
Oviedo, Spain
delacal@uniovi.es

Víctor M. González
Electrical Engineering Department
University of Oviedo
Oviedo, Spain
vmsuarez@uniovi.es

Javier Sedano
Electronic and Artificial Intelligent Unit
Instituto Tecnológico de Castilla y León
Burgos, Spain
javier.sedano@itcl.es

Samad B. Khojasteh
Department of Computer Engineering
Selcuk University
Konya, Turkey
samad.khojasteh@lisansustu.selcuk.edu.tr

***Abstract*—Fall detection has been widely studied in the literature; the main part of the studies are focused on people suffering some severe illnesses. As a consequence, the majority of the solutions relies on sensors located in positions where the ergonomic issues are in compromise. This research represents a summary of a published work concerning fall detection on healthy people using an accelerometer located on a wrist, which is more comfortable and usual gadget location. Using event detection and feature extraction, a set of variables are generated. These variables, together with on-class classifiers, conform an anomaly detection system for fall detection. The experimentation has considered two publicly available data sets and the obtained results are highly competitive. The present contribution represents a keynote about our published research.**

***Index Terms*—Fall detection, Machine Learning, One-class Classifier, Event Detection**

I. Introduction

Fall Detection (FD) refers to the detection of fall events of human beings while performing their usual Activities of Daily Living (ADL); it might be considered as part of the Human Activity Recognition (HAR). FD can be applied in several different fields, but a very interesting application field is the fall monitoring of healthy and autonomous elderly people: providing a valid FD system gives confidence and safety to the focused population, allowing an extension of the autonomous living period. This study focuses on FD using wearable devices including an tri-axial accelerometer (3DACC) placed on a wrist, which makes it easier to wear disguised in a smartwatch or bracelet; however, the complexity of the FD problem increases as so does the amount of different types of movements to consider [1], [2].

This research has been funded by the Spanish Ministry of Science and Innovation under project MINECO-TIN2017-84804-R, PID2020-112726RB-I00 and the State Research Agency (AEI, Spain) under grant agreement No RED2018-102312-T (IA-Biomed). Additionally, by the Council of Gijón through the University Institute of Industrial Technology of Asturias grant SV-21-GIJON-1-19.

FD using wearable devices has been studied for more than a decade now. Several reviews have been published and are available for an in-depth reading [3], [4]. However, there is still room for improvement when using low computational requirements models; furthermore, currently developed Deep Learning models are not valid to be deployed in the actual wearable technology without the risk of draining the batteries or completely relying on web services. This study focuses on the results in [5] and represents a keynote study.

II. A Fall Detection Method

The solution proposed includes both a generalised model and an user-centred model: the generalised model is trained with data from a large population -say, a data set of ADL and mimicked falls-, while the user-centred model is trained with data from the current user. Fig. 1 shows a block scheme of the solution: the upper part is devoted to learn generalised models, while the lower part is the solution devoted to fall detection. All the data from the sensors is standardised.

In general, the idea is to firstly detect what belongs to an abnormal performance of the subject; then, the instances are labelled as belonging to a Fall (F) or not. For the first decision we consider the use of one-class classifiers: it is difficult to obtain data from real falls but it is rather easy to obtain data from normal ADLs. Because each subject has his/her own way of behaving, these one-class models must be trained individually.

On the other hand, classifying the instances as fall or not fall relies on a two-class classifier that can be trained using the abnormal data from ADL+mimicking Fall data sets. The idea is to extract from each participant in the data set those events that are not normal and label them according to whether they were extracted or not from a Fall Time Series. Thus, this classifier is a generalised model as long as data from a large population have been used in its training.

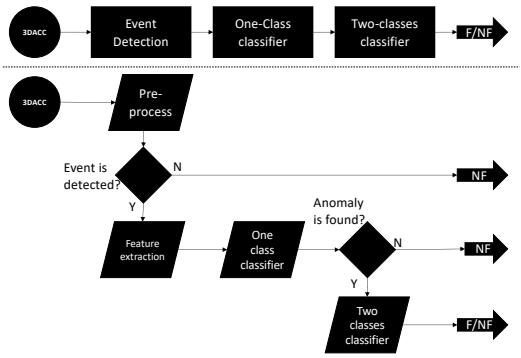


Fig. 1. Simplified block diagram of the solution. The samples are analyzed to detect the events from a suspicious behaviour. The upper part reflects the training while the lower part reflects how to deploy the solution.

The pre-processing includes a smoothing of $\frac{1}{4}$ sec plus a sliding window of 3 sec that moves every one sample. The event detection is performed using the finite state machine proposed in [6] and improved in [1]. This finite state machine receives as input the magnitude of the acceleration and outcomes whether there has been an event or not. For each detected event, a set of 8 features are computed and labelled as normal or abnormal with an one-class model; for the sake of space, refer to [5] for a complete description of these features. Finally, when the event is labelled as abnormal, then the two-class classifier decides whether it comes from a fall or not.

As one-class classifiers two possibilities have been used: an one-class Support Vector Machines and a CENTROID-based classifier. This latter model is a very simple model to obtain and deploy: the centroid and the average distance to the centroid are determined from all the detected peaks for the current user. An incoming instance is labelled as normal when it is within the hypersphere centered in the centroid and radius the average distance. On the other hand, the two-class classifier is a well-known feed forward Neural Network (TCNN) with one hidden layer. Table I shows the results for one of the possible combinations when evaluated with the Özdemir & Barshan data set. The bottom lines are the mean, median and standard deviation among the means for each participant.

TABLE I
OBTAINED RESULTS OF THE CENTROIDS+TCNN WHEN EVALUATED WITH THE ÖZDEMIR & BARSHAN DATA SET; FORWARD FALLS. SENS AND SPEC STAND FOR SENSITIVITY AND SPECIFICITY, WHILE STD IS THE STANDARD DEVIATION.

ID	FORWARD FALLS	
	Sens	Spec
	mean / std	mean / std
1	0.765 / 0.007	0.817 / 0.141
2	0.836 / 0.005	0.791 / 0.115
3	0.834 / 0.003	0.978 / 0.044
4	0.873 / 0.005	0.938 / 0.068
5	0.801 / 0.003	0.915 / 0.096
6	0.579 / 0.035	0.768 / 0.128
7	0.867 / 0.000	0.922 / 0.089
8	0.798 / 0.006	0.846 / 0.118
9	0.751 / 0.008	0.956 / 0.072
10	0.768 / 0.000	0.976 / 0.049
11	0.860 / 0.000	0.885 / 0.096
12	0.920 / 0.007	0.776 / 0.072
13	0.946 / 0.004	0.950 / 0.085
14	0.756 / 0.003	0.861 / 0.086
15	0.857 / 0.004	0.933 / 0.161
16	0.965 / 0.000	0.913 / 0.132
17	0.880 / 0.006	0.875 / 0.094
mean	**0.831**	**0.893**
median	**0.847**	**0.914**
std	**0.092**	**0.068**

III. CONCLUSION

This study proposes a three-stage process to detect fall events using a 3DACC wearable. The main contributions of this research are the event detection method and the combination of user-centred models and generalized models working together in order to detect the falls. The results show that for forward falls the one-class filtering enhances the performance of the TCNN and that specific filters are needed for other fall types.

REFERENCES

[1] S. B. Khojasteh, J. R. Villar, C. Chira, V. M. González, and E. de la Cal, "Improving fall detection using an on-wrist wearable accelerometer," *Sensors*, vol. 18, p. 1350, 2018.

[2] J. R. Villar, C. Chira, E. de la Cal, V. M.González, J. Sedano, and S. B.Khojasteh, "Autonomous on-wrist acceleration-based fall detection systems: unsolved challenges," *Neurocomputing*, vol. 452, pp. 404–413, 2021.

[3] E. Casilari-Pérez and F. García-Lagos, "A comprehensive study on the use of artificial neural networks in wearable fall detection systems," *Expert Systems with Applications*, vol. 17, p. 198, 2017.

[4] T. Mauldin, M. Canby, V. Metsis, A. Ngu, and C. Rivera, "Smartfall: A smartwatch-based fall detection system using deep learning," *Sensors*, vol. 18, no. 10, p. 3363, Oct 2018. [Online]. Available: http://dx.doi.org/10.3390/s18103363

[5] M. Fañez, J. R. Villar, E. de la Cal, V. M.González, J. Sedano, and S. B.Khojasteh, "Mixing user-centered and generalized models for fall detection," *Neurocomputing*, vol. 452, pp. 473–486, 2020.

[6] S. Abbate, M. Avvenuti, P. Corsini, J. Light, and A. Vecchio, *Wireless Sensor Networks: Application - Centric Design*. Intech, 2010, ch. Monitoring of human movements for fall detection and activities recognition in elderly care using wireless sensor network: a survey, pp. 1–22.

A platform for the design and execution of clinical data transformation and reasoning workflows

José Alberto Maldonado
Veratech for Health SL
Valencia, Spain
jamaldo@veratech.es

Mar Marcos
Universitat Jaume I
Castellón, Spain
mar.marcos@uji.es

Jesualdo Tomás Fernández-Breis
Universidad de Murcia
Murcia, Spain
jfernand@um.es

Vicente Miguel Giménez-Solano
Veratech for Health SL
Valencia, Spain
vigiso@veratech.es

María del Carmen Legaz-García
IMIB-Arrixaca
Murcia, Spain
mcarmen.legaz@ffis.es

Begoña Martínez-Salvador
Universitat Jaume I
Castellón, Spain
begona.martinez@uji.es

***Abstract*—Effective sharing and reuse of Electronic Health Records (EHRs) requires a high level of semantic interoperability. We have developed a framework to support EHR interoperability based on workflows whose primary components are reusable mappings. The framework supports mappings which take advantage of the best features of EHR standards and semantic web technologies. Based on this framework, we have implemented CLIN-IK-LINKS, a web-based platform that enables users to create, modify and delete mappings, as well as to define and execute clinical data transformation and reasoning workflows. The platform has been applied in two use cases in which real data have been used. This *key work* is an abridged version of the paper published in the journal Comput Meth Prog Bio [1].**

***Index Terms*—Data Workflow, Electronic Health Records, Health Information Interoperability, Semantic Web**

I. Introduction

Electronic Health Record (EHR) systems promised a series of benefits including: savings from data capture and access; information connectivity for stakeholders; and improvement of efficiency, safety, and quality of healthcare through the use of decision support. Some studies have shown that such benefits are still far from being achieved due to the difficulties of sharing and reusing EHR content. Effective EHR sharing and reuse requires a high level of semantic interoperability of EHR data. To achieve this, leading international initiatives and projects have recommended the use of EHR standards and semantic resources.

In the last years we have explored how EHR standards and semantic web technologies could be combined to overcome data heterogeneity in a patient phenotyping scenario [2]. One lesson learned was the need to apply a number of data transformations of different types, and that the organisation of the transformations in a workflow could promote the reuse of the data models and the transformations. This led us to explore how web services could be used to implement such workflows. Another lesson learned was the need to precisely characterise the mappings underlying the data transformations to facilitate reuse.

Based on our previous work, in this paper we describe: (1) a conceptual framework for representing clinical data transformation and reasoning services, (2) the CLIN-IK-LINKS platform that we have implemented to support such framework, and (3) a series of EHR-related interoperability scenarios demonstrating its usefulness.

Work supported by the Spanish Ministry of Economy and Competitiveness and the ERDF programme through grants TIN2014-53749-C2-1-R, TIN2014-53749-C2-2-R, and TIN2017-85949-C2-1-R, and by the Spanish Ministry of Science, Innovation and Universities through grant PTQ2018-009924.

II. Material & Methods

CLIN-IK-LINKS is aimed at facilitating the composition and execution of clinical data transformation workflows to convert EHR data stored in a database into EHR and/or semantic web standards. While focused on data transformation workflows, the platform relies upon a collection of reusable *mappings* with associated *web services*. Mappings are declarative descriptions that specify the operations to be performed to obtain data conforming to a target data model from data stored according to a source data model, designed for specific transformation steps in a clinical scenario. Web services are the essential complement of mappings. Depending on the transformation task they perform, they can be categorised into: normalization, abstraction, semantic publishing and reasoning.

The platform is based on a conceptual framework allowing for the characterisation of mappings beyond the data/information artefacts they consume/produce, At the information model level, the framework distinguishes between the executable mappings (*Mapping*) and the entities that can be mapped with them (*MappableEntity*). The latter includes clinical information models or artefacts in various forms, e.g XML schemas. At this level, a mapping is related to one or more input artefacts and to one output artefact. At the concept level, conceptual mappings (*ConceptualMapping*) and concepts (*Concept*) are the counterpart of mappings and mappable entities, respectively. Using these elements, one can specify not only the connection between a mapping implementation and its definition (via the *IMPLEMENTS* relationship), but

also the relation between the mapping definition and its input and output concepts. Additionally, the connection between an information artefact and the clinical concept it represents can be explicitly stated (with the *REPRESENTS* relationship).

Using the above elements, the platform can accommodate a variety of cases, e.g. several mappings implementing the same conceptual mapping, based or not on the same artefacts, or even multiple conceptual mappings that serve as alternative methods to derive a particular concept.

III. Results

A. The platform

The CLIN-IK-LINKS platform has been implemented in J2EE and uses Neo4J (a graph database) for managing artefacts, concepts and mappings as well as the relationships among them. The overall architecture is diagrammed in Fig. 1. The workflow engine is implemented as a service that offers a REST API. Users interact with the platform by means of a web-based front-end application that allows them to create, modify and delete artefacts and their relationships, as well as to define and execute workflows.

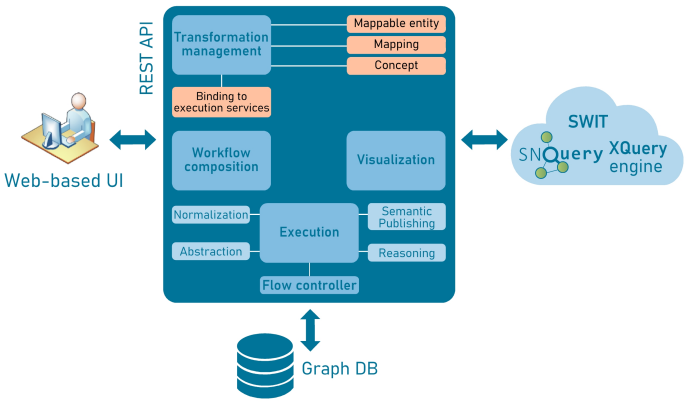


Fig. 1. Overall architecture of the CLIN-IK-LINKS platform.

The set of mappings, artefacts, concepts, and relations among them yields as a result a (disconnected directed acyclic) graph. In the platform, a workflow is defined through a query over this graph. Given either a source or a target artefact, or both, the platform offers a number of paths defining transformation workflows from the source to the target, made up of one or more mappings. For querying purposes, the visual interface presents the user with the set of paths between the source and target artefacts. Such visualisation has proven very useful in composing and understanding complex workflows. The set of paths (or possible transformation workflows) is calculated using the graph traversal capabilities of the Neo4J query language.

B. Use cases

We have implemented two different use cases, available at http://keg.act.uji.es/demoCLINIKLINKS/, which operate with real data. The first use case transforms clinical laboratory test results into an OWL representation in terms of a LOINC-based ontology, and the second one implements two different colorectal cancer screening protocols. In both use cases, we have used existing tools for the definition of mappings and the generation of data transformation programs, namely LinkEHR and SWIT. Note that any mapping tool providing an appropriate web service could have been used instead. In addition, both use cases required elements such as database schemas, clinical information models (e.g. archetypes), and ontologies. These items had to be developed specifically for the use case when they were not available.

IV. Discussion

CLIN-IK-LINKS provides an integrated representation, storage and exploitation of the different sorts of mappings created in the platform. This not only promotes reuse and collaboration among mapping designers, but also facilitates the discovery of mapping compositions/workflows not initially foreseen. In addition, the platform provides flexibility in the design of workflows, since they are composed from queries to the graph of mappings. This allows the user to analyse the different options, and to select the optimal one according to the actual needs.

Several widely used workflow systems are available in biomedical informatics, such as Taverna or Galaxy. However, according to our experience such systems do not provide the necessary features to model workflows with the specific requirements of the healthcare domain, where e.g. there is an increasing number of sophisticated terminological resources. Our platform includes a concept-level layer that not only allows you to describe the relationships between mapping definitions and their input and output concepts, but also to link concepts with appropriate terms from existing terminologies. On the other hand, state-of-the-art workflow systems are oriented towards the definition and exploitation of individual workflows, whereas CLIN-IK-LINKS leverages a graph of mappings which integrates content from all the workflows defined in the platform, thus facilitating sharing and reuse.

V. Conclusions

In this paper we have described CLIN-IK-LINKS, a platform for the composition and execution of clinical data transformation workflows to convert EHR data into EHR and/or semantic web standards. Although the platform was originally conceived to design EHR data transformation workflows for interoperability purposes, we have shown that it can be used to implement applications of a certain significance by themselves, e.g. normalisation and semantic publishing of EHR data.

References

[1] J A Maldonado, M Marcos, J T Fernández-Breis, et al. CLIN-IK-LINKS: A platform for the design and execution of clinical data transformation and reasoning workflows. *Comput Meth Prog Bio*, 197:105616, 2020.

[2] J T Fernández-Breis, J A Maldonado, M Marcos, et al. Leveraging electronic healthcare record standards and semantic web technologies for the identification of patient cohorts. *J Am Med Inform Assoc*, 20(e2):e288–e296, 2013.

DISNET: extracting public phenotypical knowledge of diseases

Lucía Prieto Santamaría
Centro de Tecnología Biomédica, ETS Ingenieros Informáticos
Universidad Politécnica de Madrid
Madrid, Spain
lucia.prieto.santamaria@upm.es
ORCID: 0000-0003-1545-3515

Ernestina Menasalvas Ruiz
Centro de Tecnología Biomédica, ETS Ingenieros Informáticos
Universidad Politécnica de Madrid
Madrid, Spain
ernestina.menasalvas@upm.es
ORCID: 0000-0002-5615-6798

Alejandro Rodríguez-González
Centro de Tecnología Biomédica, ETS Ingenieros Informáticos
Universidad Politécnica de Madrid
Madrid, Spain
alejandro.rg@upm.es
ORCID: 0000-0001-8801-4762

***Abstract*—In this paper we introduce DISNET (http://disnet.ctb.upm.es/), a web-based platform that integrates phenotypical knowledge of diseases extracted from public textual sources. Such knowledge includes signs and symptoms associated to diseases that are extracted periodically, providing a traceability in time. Moreover, DISNET enables the creation of customizable disease networks, allowing researchers to go deep in the understanding of diseases and the connections stablished between them.**

***Keywords*—DISNET, knowledge base, disease understanding, disease symptoms, data mining, natural language processing, phenotypic information, public sources**

I. Introduction

Understanding diseases and their relationships is an arduous but relevant task, needed (among other purposes) to better treat each condition. In these modern times, the presence of data and its manipulation are crucial in each domain, including life sciences and medicine. Data is being used in the pursue of such a better grasp of the causes and effects of diseases, unveiling uncovered associations and insights underpinning biology. The identification and integration of medical knowledge spread through several information sources is key to accomplish an improvement in population health.

One of the significant aspects of a disease is the set of symptoms related to it. The phenotypical facet exposed by a disease gives interesting information of its nature and underlying processes, and should thereby be considered when integrating biomedical knowledge. In this context, the creation of a comprehensive dataset of diseases and their clinical manifestations allows not only to complement and merge medical knowledge but also to increase it, interconnecting existing data and improving the analysis of the relationships of diseases.

The current manuscript derives from a work [1] published in PeerJ scientific journal, and aims to present DISNET (http://disnet.ctb.upm.es/), a web-based system designed to periodically extract the knowledge from diseases' signs and symptoms retrieved from mining public textual sources. Here, we summarize the main contributions of DISNET as a public platform to provide disease phenotypic data, but for further information, readers are referred to the original article.

II. Methods

Some of the aspects presented in this section include (i) the sources providing medical phenotypic information and how DISNET system retrieves the texts from the named sources, (ii) how these texts are mined to extract diagnosis terms, and (iii) how such terms are validated to compile a final list of valid "symptom-type" terms.

Bringing together information from very different sources can complement each of them, providing more comprehensive relationships when building new disease networks. Wikipedia (English edition) and PubMed were selected as the information sources to extract disease phenotypic knowledge, since they are written in very different styles and by people with very different backgrounds. However, DISNET users are able to later choose whether they want to query the data from one or multiple sources or applying several prevalence or other kinds of filters.

DISNET architecture is presented in **Figure 1** and summarized as follows. We first extract the texts that are considered as a disease from each source. To determine which documents are about a disease, we build a list of diseases. Once determined this disease list, DISNET system mines "sign and symptoms" section from Wikipedia for each disease document; and, from PubMed, it mines the abstract section of each publication related to the named disease. Wikipedia is mined twice a month since February 2018, whilst PubMed has just been processed once (April 2018).

The process to obtain terms from the texts follows a Natural Language Processing (NLP) pipeline, and it is implemented by MetaMap tool [2]. Specific configuration details of such tool are included in DISNET web. After MetaMap mining the texts, the medical terms obtained, representing phenotypical manifestation concepts, are filtered and validated by TVP process. All the information is stored in a relational database instance that can be externally queried by means of an API REST (http://disnet.ctb.upm.es/apis/disnet).

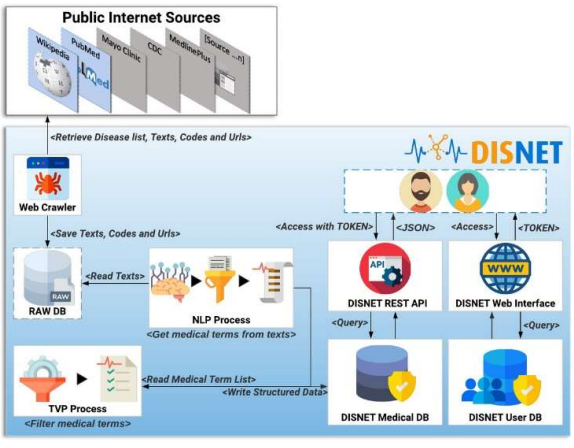


Figure 1. DISNET Architecture/Workflow.

XXX-X-XXXX-XXXX-X/XX/$XX.00 ©20XX IEEE

III. Results and discussion

At the time the referred paper was published, the database in DISNET system contained 26 snapshots from Wikipedia (it ascends to 80 in the current time, May 2021) and one from PubMed. All the aforementioned data can be retrieved through the system's API and supplementary information about the dataset is provided in the web.

In order to obtain accuracy metrics' values and to have an evaluation pipeline for our system, some parts of the information extracted were manually annotated by 3 people as True Positives, False Positives, True Negatives and False Negatives. The validation for Wikipedia was carried out on February the 1st, 2018 snapshot, selecting 100 diseases at random with the only condition of having at least 20 valid medical terms in order to ensure that our evaluation procedure analysed articles with a high concentration of medical knowledge. Similarly, the validation for PubMed has been done selecting a random sample of 100 article abstracts. For Wikipedia, to validate an article means to validate a disease, for PubMed to validate an article means to validate a part of a disease.

Results indicated that the followed NLP pipeline was sufficiently reliable, with an accuracy of 0.731 (confidence interval of [0.710, 0.753], calculated through a Wilson's score interval with continuity correction and a confidence level of 99%) for Wikipedia and of 0.640 (confidence interval of: [0.606, 0.680]) for PubMed. It is noteworthy the large difference of false positive rates between Wikipedia (11.41%) and PubMed (17.54%). We speculated this was due to the correction of articles. In Wikipedia, articles corresponding to one disease refer almost exclusively to that particular disease, and thus include no irrelevant terms – with a few exceptions related to differential diagnoses. However, a significant part of PubMed articles are not so specific: the same PubMed document includes symptoms of many different diseases that, although being true medical terms and thus being recognized by MetaMap, may not be relevant to the disease under analysis.

The final output of this work includes the creation of a comprehensive symptoms-disease dataset, shared (free access) through the system's API. In the original paper, some examples of the returned data and its possible visualizations were deeply analysed. The API has the capacity to create a variety of queries representing DISNET information, being two of the most important characteristics of the system the following two: i) the ability to create relationships between diseases according to their phenotypic traits and ii) the ability to increase/improve the phenotypic information of diseases by means of periodic extractions of knowledge.

IV. Conclusions

DISNET allows retrieving knowledge about the signs, symptoms and diagnostic tests associated with a disease. It is not limited to a specific category (all the categories that the selected sources of information offer us) and clinical diagnosis terms. It further allows to track the evolution of those terms through time, being thus an opportunity to analyse and observe the progress of human knowledge on diseases. The validation pipeline has suggested that DISNET is good enough to be used to extract diseases and diagnostically-relevant terms. At the same time, the evaluation also revealed that improvements could be introduced to enhance the system's reliability. Finally, DISNET is characterized by a high flexibility, such that new information sources can easily be included (provided they contain the appropriate type of information). DISNET also includes a REST API, which aims at sharing the retrieved information with the wide scientific community.

There is a considerable amount of possibilities in future works. The number of information sources could be increased: complementary sources would include Medline Plus or CDC, among others. The TVP procedure could be extended, and other configurations of MetaMap could be explored. Future implementations of DISNET aim to provide methods to automatically compute similarity metrics between diseases and cover biological and drug information. The semantization of the complete dataset would improve data integration and interoperability and could give new possibilities in the context of SPARQL queries.

Acknowledgments

The work is a result of the project "DISNET (Creation and analysis of disease networks for drug repurposing from heterogeneous data sources applied to rare diseases)", that is being developed under grant "RTI2018-094576-A-I00" from the Spanish Ministerio de Ciencia, Innovación y Universidades. LPS's work is supported by "Programa de fomento de la investigación y la innovación (Doctorados Industriales)" from Comunidad de Madrid (grant IND2019/TIC-17159).

References

[1] G. Lagunes-García, A. Rodríguez-González, L. Prieto-Santamaría, E. P. G. del Valle, M. Zanin, and E. Menasalvas-Ruiz, 'DISNET: a framework for extracting phenotypic disease information from public sources', *PeerJ*, vol. 8, p. e8580, Feb. 2020, doi: 10.7717/peerj.8580.

[2] 'MetaMap - A Tool For Recognizing UMLS Concepts in Text'. https://metamap.nlm.nih.gov/ (accessed May 26, 2021).

Using Machine Learning for the prediction of mutations in the initiation codon

Javier Castell-Díaz
Universidad de Murcia
Murcia, Spain
javier.castelld@um.es

Francisco Abad-Navarro
Universidad de Murcia
Murcia, Spain
francisco.abad@um.es

Jesualdo T. Fernández-Breis
Universidad de Murcia
Murcia, Spain
jfernand@um.es

M.Eugenia de la Morena Barrio
Universidad de Murcia, IMIB-Arrixaca
Murcia, Spain
mml13317@um.es

Javier Corral de la Calle
Universidad de Murcia, IMIB-Arrixaca
Murcia, Spain
jcc@um.es

***Abstract*—The study of mutations in DNA is important since they mean changes with respect to the reference genome. Some mutations may have an impact on the health of humans, therefore, being able of predicting which mutations may be harmful becomes a scientific challenge. In the last decades, techniques such as SIFT or PolyPhen apply a general model for predicting the effect of an amino acid change produced by a mutation. However, such general models do not reflect the biology of all the parts of the genome, such as the biology related to the initiation codon. In this work, we have studied how to develop a predictor for the effect of mutations in the initiation codon by using machine learning techniques, based on the biologic model proposed in prior work, focusing on the distance to the alternative initiation codon and the status of the original reading frame. We worked with real data from *Ensembl* and obtained a classifier by combining different tree-based techniques that has a *ROC_AUC* of 63.2% and an accuracy of 82.2%, outperforming SIFT and PolyPhen for this type of mutation. All the code and data used is available at https://github.com/JavierCastellD/PredictorMutacionCodonInicio.**

***Index Terms*—machine learning, initiation codon, mutation, imbalance**

I. Introduction

DNA is in the nucleus of all cells and contains the sequence of information that is needed to create proteins. Those proteins are made outside the nucleus, in the ribosomes, so the information needs to be transported by the RNA through a process called transcription, where the needed information is copied from the DNA to the RNA. When the transcript arrives to the ribosome, the sequence is scanned until an AUG codon in a favorable context is detected, so that the translation to proteins can start.

Mutations are changes that occur in the DNA due to errors during the transcription or translation or because of environmental factors. Those mutations mean a change in the bases that conform DNA and can affect interactions between proteins [16], [22] or the proteins' functions itself [24], [26].

In this work we focus on mutations that occur in the initiation codon, as a change in that point in the sequence can mean that a different protein will be produced. There has been previous work done predicting the effects of mutations, such as *SIFT* [19] or *PolyPhen* [2], that use the levels of amino acid conservation and the effect that a substitution of one amino acid could have. However, those methods are not adequate for mutations in the initiation codon, since this kind of mutation is not necessarily related to the functions of the new protein that might be generated, as it may prevent the translation to protein to happen, due to not being able to find a starting site, and therefore there would not be an amino acid change [1].

This is why we are focusing on developing a predictor of the effect of mutations that affect the initiation codon in the transcription of human genome using machine learning algorithms. There has been an increase in the use of machine learning techniques during last years due to recent advancements in sequencing techniques that have lead to more data being available for the training of those models [20], [28].

Our work applies the hypothesis presented in [1]. If the original initiation codon is lost, the translation may begin in the next initiation codon that is in an adequate context. Therefore, we should take into account variables such as the distance to the alternative initiation codon from the original and whether the original reading frame is maintained or not, as the farther the distance or if the reading frame is not maintained, the greater the chances are that the mutation is harmful for the individual.

This document is organized as follows: in section II we explain the methodology applied and the dataset used, in section III we show our results and discuss about them, and finally, in section IV, we talk about future work on this subject.

II. Materials and Methods

We proposed a method based on machine learning techniques to obtain a model that is capable of classifying mutations in the initiation codon as benign or deleterious.

Our method is shown in figure 1 and has three main steps, which are further described in the next subsections:

- Application of machine learning algorithms, such as Random Forest [4] or SVM [8], and different general parameters like the number of variables of our dataset to use, as well as the percentage of undersampling.
- Hyperparameter fine-tuning of those algorithms chosen in the previous step trying to prevent overfitting.
- Creating a voting ensemble of the best performing models obtained in the second step.

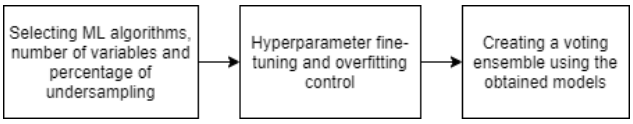


Fig. 1. Process followed to obtain the predictor.

A. The dataset

The dataset used in this work is formed by the 26037 mutations observed in the initiation codon in the human genome (*GRCh38* version), and 199 mutations from the goat genome (*ARS1* version). We used goat genome to increase the number of instances of benign mutations and it could be used in combination with the human instances since the transcription and translation genetic mechanism is the same in both organisms. This necessity came from the extremely imbalance of our dataset, as 98.5% of the instances (25814) belonged to the *DELETERIOUS* class, while the remaining 1.5% (422) corresponded to the *BENIGN* class. 36% of those 422 *BENIGN* instances come from the goat genome.

We obtained this dataset from *Ensembl* [27] and it contains the target feature CLASS and 13 features, six of which are categorical and, the rest, numerical. The distribution of each variable and what it represents can be seen in tables I and II.

At the beginning of our work, we split our dataset in two subsets: training and a test, with each containing 90% and 10% of the original dataset respectively. The split was done in a stratified fashion, so that each subset contains approximately the same percentage of samples of each target class as the complete set. The training subset is the one we used during all of our experiments and in several instances, it was split into training and evaluation subsets. However, the test subset was only used at the end of the experiment to obtain the final results for our model.

B. Algorithm, variables and undersampling selection

The objective of this phase is to select some machine learning algorithms, as well as which combination of variables and percentage of undersampling we should use.

When performing feature selection, the number of variables was determined by using a combination of five statistical tests: chi-square, ANOVA, mutual information and LASSO logistic regression with liblinear and saga. This was done in order to reduce bias, as selecting one over the other depends on the training set [3] and it is based on the one used in [9]. We developed two very similar techniques (see figure 2) that both required to ask a number of features to each statistical test. In the first technique, we asked for n features to choose the n ones most selected by our tests. However, for the second technique, we chose features selected by at least one test.

Besides choosing the number of features, in this step we also selected the percentage of undersampling to apply, which affects to how much of the training set corresponds to the minority class *BENIGN*, as well as whether to use cost sensitive learning, so that the weight of the data is influenced by the frequency of its class. Several supervised machine learning algorithms were tested: *SVM*, *KNeighbours*, *AdaBoost* [10], *Gradient Boosting* [11], *GaussianNB*, *Gradient Descent* [17], *Decision Tree* [6], *Random Forest*, *Extremely Randomized Trees* [13] and *Bagging Classifier* [5].

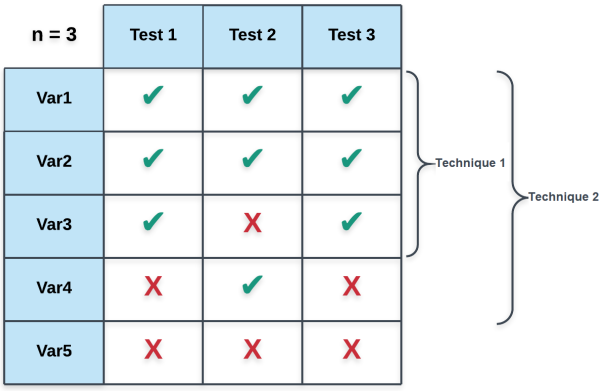


Fig. 2. Feature selection techniques.

The output of this step was a group of different models elected by training a combination of machine learning algorithms, number of features, feature selection techniques, percentage of undersampling, and cost sensitive learning, and then choosing the most promising ones. We consider as promising in this phase a model with a ROC_AUC greater than 0.6, an accuracy greater than 0.75, and a specificity greater than 0.45. We did this in four batches (each one being a combination of cost sensitive usage and feature selection technique) and chose the top three of each batch after sorting them by ROC_AUC, as it is one of the main metrics for unbalanced classes [7], [12].

C. Hyperparameter fine-tuning

The objective of this step is to fine-tune the hyperparameters of the models chosen in the previous section. In order to do so, we selected different values for its hyperparameters and performed an exhaustive search, trying every combination for the values we selected with the objective of maximizing the ROC_AUC value.

For each combination, we perform stratified cross-validation to preserve the percentage of each class and to avoid training with elements of only the majority class. As we want to prevent overfitting and this is likely to happen for the *BENIGN* class, we will only chose those who have a difference of less than 0.35 between training and test in the specificity value.

TABLE I
CATEGORICAL FEATURES

FEATURE	MEANING
CLASS	Type of mutation.
CDS_COORDS	Mutation coordinates in relation to the coding region.
AMINOACID_CHANGE	Type of aminoacid change.
CODON_CHANGE	Type of codon change.
READING_FRAME_STATUS	Whether the reading frame has been maintained.
NO_STOP_CODON	Whether there is not a stop codon.
PREMATURE_STOP_CODON	Whether there is a premature stop codon.

TABLE II
NUMERICAL FEATURES

FEATURE	MEANING
NMETS_5_UTR	Number of ATG in the 5' UTR.
CONSERVED_METS_IN_5_UTR	Number of maintained ATG in the 5' UTR.
LOST_METS_IN_5_UTR	Number of lost ATG in the 5' UTR.
CONSERVED_METS_NO_STOP_IN_5_UTR	Number of maintained ATG in the 5' UTR without stop codon.
MET_POSITION	Distance to the alternative initation codon.
STOP_CODON_POSITION	Distance to the stop codon.
MUTATED_SEQUENCE_LENGTH	Length percentage of the mutated aminoacid.

We also established a minimum of 0.65, 0.4 and 0.6 for the accuracy, specificity and ROC_AUC values respectively.

D. Voting ensemble

From the optimized models of the previous phase, we chose the top five according to the ROC_AUC metric and created voting ensembles that consisted of each combination of three different models as well as one with the five ones. During this procedure, we also checked whether to use a *hard* or *soft* voting, so that the predicted class was determined by a majority rule or by the maximum of the sum of the predicted probabilities of each class. The final classifier was the one with the most ROC_AUC and with a minimum specificity of 0.4.

E. Comparison with SIFT and PolyPhen

To compare the performance of our method with state of the art methods such as PolyPhen [2] or SIFT [19], we used a version of the whole dataset that only contains mutations of the initiation codon of human genome, as PolyPhen is focused on human genome, and the list of genomes for SIFT does not contain the goat (*capra hircus*) [18].

Therefore, we divide this version of the dataset in training and test sets so that we train our model with the former, and compare the predictions of both our model and the others with the latter. We repeated this process a thousand times to reduce the variance under 1%.

III. RESULTS AND DISCUSSION

In the first phase of our experiment we discarded almost all machine learning algorithms but the tree-based ones, as *Random Forest* (*RF*), *Decision Tree* (*DT* and *BCDT*) and *Extremely Randomized Forest* (*ET*) were the ones with the most prominent results independent of the use of cost sensitive learning or feature selection technique, which were used to differentiate each parameter training batch. In tables III and IV we can see the parameters for the top three performing models of each batch.

The values we tried for the random undersampling ([0.05, 0.1, 0.2, 0.3, 0.4, 0.5]) were between 5% and 50%, which would make our dataset completely balanced at the cost of losing almost all of our samples, while the number of features was tested between 2 and 5 for both feature selection techniques, since during a previous analysis of our dataset we found out that five features had a significance difference in their p-values from the others and at least eight had a significant relation with our target class (p-value of less than 0.05). The use of thresholds to filter out some of the models was done so that we could have a mechanism to avoid a model which only predicts the majority class, as even using ROC_AUC as the main metric we can find models that have similar ROC_AUC values than our models while also having difficulties predicting the minority class, obtaining results of around 0.3 of specificity or less.

By comparing both feature selection techniques we could see that the number of variables in the second technique tends to be less, as it does not necessarily mean that only two or three features would be selected. The second technique was also more prone to overfitting, as we will see below, as the number of features rose, so did the complexity of the model.

During the second phase, we found that while performing hyperparameter optimization, a considerable number of configurations were prone to overfitting for the specificity metric, probably due to the low number of available instances during training. In those cases we found values close to 1 for specificity during training and less than 0.3 during testing, which made us establish the threshold of a maximum of 0.35 difference in order to accept a model. We also reduced the thresholds for accuracy and specificity, as the number of instances for training during this phase was less due to the way we performed the exhaustive research, which was done

TABLE III
PHASE 1 RESULTS FOR FEATURE SELECTION TECHNIQUE 1. *US* STANDS FOR PERCENTAGE OF UNDERSAMPLING, *N_var* FOR NUMBER OF FEATURES FOR THE FEATURE SELECTION, *CS* FOR USE OF COST SENSITIVE LEARNING, AND *Acc* AND *Spec* FOR ACCURACY AND SPECIFICITY RESPECTIVELY

ID	US	N_var	CS	Acc	Spec	ROC_AUC
DT6	4	0.2	YES	0.830	0.478	0.657
DT7	5	0.2	YES	0.829	0.465	0.650
RF2	4	0.4	YES	0.802	0.490	0.649
DT1	5	0.3	NO	0.809	0.476	0.645
DT2	4	0.4	NO	0.761	0.521	0.643
ET1	4	0.4	NO	0.812	0.466	0.642

TABLE IV
PHASE 1 RESULTS FOR FEATURE SELECTION TECHNIQUE 2

ID	US	N_var	CS	Acc	Spec	ROC_AUC
ET2	2	0.4	NO	0.860	0.486	0.676
RF1	2	0.4	NO	0.870	0.472	0.674
BCDT1	2	0.4	NO	0.802	0.523	0.665
DT3	2	0.2	YES	0.860	0.458	0.662
DT4	3	0.2	YES	0.865	0.451	0.661
DT5	3	0.3	YES	0.810	0.497	0.656

by using *GridSearchCV* from *scikit-learn* library [21].

The values we tried for the hyperparameters of those twelve models from the previous step were the following: *max_depth* with values in [1, 2, 4, 8, 16, 32, 64, 128], *min_samples_split* with values in [2, 5, 10], *min_samples_leaf* with values in [1, 2, 4], *n_estimators* with values in [1, 2, 5, 10, 20, 30, 40, 50, 100, 200, 300, 400, 500], *bootstrap* with values in [*True*, *False*] and *criterion* with values in [*'gini'*, *'entropy'*]. We can see the results of the top five performing models of this phase in table V, which were then used as the baseline models for our voting ensembles. Those results were obtained after repeating the training and testing a hundred times in order to reduce variance.

In the last step, we decided to create a voting ensemble, to further reduce overfitting. When creating them, we tested each combination of three different models from the top five of step two, as well as one that combines all five, and similar to what we did for phase two, we repeated the process of training and evaluating by creating different splits so that we could reduce variance and make a better decision. We also had to decide whether to use *hard* or *soft* voting, as well as which feature selection technique was the most appropriate, so as we did in phase one, we created four batches and chose the three best models of each to then compare them. However, this time we chose the two top models according to ROC_AUC and the most promising one according to ROC_AUC that also has specificity over 0.5 instead of a threshold. Of the eleven possible combinations, only six of them had the best results and they can be seen in table VI.

The results obtained for those models can be seen in tables VII and VIII and the one highlighted was the selected model, as it had the highest ROC_AUC value, and had over 0.5 in specificity. We can see an increase in performance in ROC_AUC thanks to the use of voting ensembles. There is a 30% difference in comparison with the results of the models of the previous step.

The final performance results for the *VC11B* model were done by using the whole training set for training and then testing said model with the evaluation subset we set apart at the beginning. We repeated this experiment of training and testing a hundred times in order to have a variance of under 1%, but we did not change the composition of said subsets. The results can be seen in table X with a confidence interval of 95%. The parameters for the *VC11B* model can be seen in table IX, while the hyperparameters for its baseline models can be found in table V.

Finally, we compared the performance of our model with PolyPhen and SIFT. As we previously mentioned, we used a different version of our dataset that only contained mutations of the initiation codon for human genome, as our data for goat genome did not have the needed features to perform the prediction for PolyPhen and SIFT. The results can be seen in table XI with a confidence interval of 95% after repeating the training and testing a thousand times.

Our model performed better than SIFT in the prediction of both *BENIGN* and *DELETERIOUS* mutations, as seen in the difference in specificity 64.34% vs 34.83%, and recall 84% vs 82.70% respectively. When it comes to PolyPhen, our model performed better predicting *DELETERIOUS* mutations, 84% vs 54.15%, but falls behind when predicting *BENIGN* mutations. However, we consider that *VC11B* is preferable over PolyPhen, since PolyPhen suffers from a low recall, which means that it has problems with false negatives, while our model has a problem of false positives. In this field it is better to have false positives, diagnosing that a mutation is deleterious when, in reality, it is benign, than false negatives, which is the other way around: diagnosing a mutation as benign when it is deleterious. This is because if we wrongly classify a mutation as deleterious, the false positive can be detected with further medical tests.

Additionally, it is also interesting that our model had a better performance in the dataset with only human data, an improvement that it is specially noted in the prediction of the *BENIGN* class. This might mean that by adding goat genome data, we are actually adding unnecessary noise, or that the benign mutations added by goat genome made it harder for the model to generalise. It is also possible that, due to the lower number of samples of the *BENIGN* class, our model is able to better learn them.

IV. CONCLUSION

We have obtained a predictor capable of detecting whether a mutation in the initiation codon will be deleterious or benign by using a voting ensemble composed of several tree-based machine learning algorithms. This model, which we called *VC11B* during testing, achieved a performance of 63.2% *ROC_AUC*, as well as an accuracy of 82.2%, and an specificity of 43.7%, because it has limitations to identify *BENIGN* mutations and miss-labels some of them as *DELETERIOUS*.

TABLE V
PHASE 2 RESULTS

ID	max_depth	min_samples_split	min_samples_leaf	n_estimators	bootstrap	criterion	Acc	Spec	ROC_AUC
DT5	64	2	4	-	-	gini	0.677	0.584	0.631
ET1	64	2	1	300	True	-	0.843	0.399	0.625
RF2	32	2	2	10	False	-	0.693	0.552	0.623
BCDT1	16	5	2	10	False	-	0.772	0.465	0.621
DT6	16	2	2	-	-	gini	0.707	0.516	0.613

TABLE VI
BASELINES MODELS FOR EACH VOTING ENSEMBLE

ID	Baseline models
VC1	DT5, ET1, RF2
VC3	DT5, ET1, DT6
VC4	DT5, RF2, BCDT1
VC5	DT5, RF2, DT6
VC8	ET1, RF2, DT6
VC11	DT5, ET1 RF2, BCDT1, DT6

TABLE VII
PHASE 3 RESULTS FOR FEATURE SELECTION TECHNIQUE 1

ID	US	N_var	Voting	Acc	Spec	ROC_AUC
VC5A	0.05	4	hard	0.841	0.455	0.651
VC3	0.05	4	hard	0.849	0.423	0.639
VC8	0.3	5	soft	0.829	0.437	0.636
VC11A	0.2	3	hard	0.762	0.493	0.630
VC1A	0.3	5	soft	0.813	0.442	0.630

TABLE VIII
PHASE 3 RESULTS FOR FEATURE SELECTION TECHNIQUE 2

ID	US	N_var	Voting	Acc	Spec	ROC_AUC
VC4	0.1	2	hard	0.929	0.397	0.667
VC1C	0.1	2	soft	0.941	0.383	0.667
VC11B	**0.4**	**3**	**hard**	**0.809**	**0.515**	**0.665**
VC1B	0.05	2	soft	0.965	0.355	0.665
VC5B	0.1	2	hard	0.875	0.447	0.665
VC1D	0.4	2	soft	0.799	0.498	0.651

TABLE IX
PARAMETERS FOR *VC11B*

Base models	DT5, ET1, RF2, BCDT1, DT6
Undersampling	40%
Number variables	3
FS Technique	2
Voting	Hard

TABLE X
FINAL RESULTS FOR *VC11B*

ID	Accuracy	Specificity	ROC_AUC
VC11B	82.21 ± 0.29	43.67 ± 0.95	63.25 ± 0.50

TABLE XI
COMPARISON WITH POLYPHEN AND SIFT

ID	Accuracy	Specificity	Recall
SIFT	82.27 ± 0.03	34.83 ± 0.38	82.70 ± 0.03
PolyPhen	54.36 ± 0.04	76.28 ± 0.33	54.15 ± 0.04
VC11B	83.81 ± 0.11	64.34 ± 0.44	84 ± 0.11

However, when comparing it with others such as PolyPhen or SIFT, it ends up performing better for this type of mutations.

According to the variables that were selected by our feature selection techniques, we can also say that the hypothesis of [1] holds, since both the distance to the alternative initiation codon (MET_POSITION) and the status of the reading frame (READING_FRAME_STATUS) were chosen.

For future work, having more data, or at least a less imbalanced dataset with more instances of the *BENIGN* class, may improve the quality of the predictor, as this has been one of the main challenges of this work. It might be interesting to try different techniques to deal with the imbalance such as a one class classifier [14] or other alternatives based on the use of neural networks [15], [25], as well as testing other feature selection techniques, such as multivariate feature selection.

ACKNOWLEDGMENT

This work has been possible thanks to the the Ministerio de Ciencia, Innovación y Universidades, Agencia Española de Investigación and the ERDF programme through grant TIN2017-85949-C2-1-R.

REFERENCES

[1] Abad-Navarro F, Morena-Barrio M, Fernández-Breis J, Corral J. "Lost in Translation: Bioinformatic Analysis of Variations Affecting the Translation Initiation Codon in the Human Genome" *Bioinformatics.* 2018. doi:10.1093/bioinformatics/bty453

[2] Adzhubei IA et al. "A method and server for predicting damaging missense mutations." *Nat Methods.* 7(4), 248-9. 2010. doi:10.1038/nmeth0410-248

[3] Aldehim, G., Wang, W. "Determining appropriate approaches for using data in feature selection" *Int. J. Mach. Learn. & Cyber.* 8, 915–928. 2017. doi:10.1007/s13042-015-0469-8

[4] Breiman L. "Random Forests" *Machine Learning.* 45, 5-32. 2001. doi:10.1023/A:1010933404324

[5] Breiman L. "Bagging predictors" *Machine Learning.* 24, 123-140. 1996. doi:10.1023/A:1018054314350

[6] Breiman L, Friedman J, Olshen R, Stone C. "Classification and Regression Trees" 1983.

[7] Brzezinski D, Stefanowski J. "Prequential auc for classifier evaluation and drift detection in evolving data streams" *Lectures Notes in Computer Science* 2015. doi:10.1007/978-3-319-17876-9_6

[8] Cortes C, Vapnik V. "Support-vector networks" *Machine Learning.* 20, 273-297. 1995. doi:10.1007/BF00994018

[9] Dubey R, Zhou J, Wang Y, Thompson P, Ye J. "Analysis of sampling techniques for imbalanced data: An n = 648 ADNI study" *NeuroImage.* 87, 220-241. 2014. doi:10.1016/j.neuroimage.2013.10.005

[10] Freund Y, Schapire R. "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting" *J. Comput. Syst. Sci.* 55, 119-139. 1997. doi:10.1006/jcss.1997.1504

[11] Friedman J. "Stochastic Gradient Boosting" *Computational Statistics and Data Analysis.* 38, 367-378. 2002. doi:10.1016/S0167-9473(01)00065-2

[12] García Ruiz S. "Una aproximación al problema del desbalanceo en el contexto del aprendizaje automático: aplicación al diagnóstico genético de discapacidad intelectual" [An approximation to the imbalance problem in the context of machine learning: application to the genetic diagnostic of intellectual disability] TFM Universidad de Murcia. 2018.
[13] Geurts P, Ernst D, Wehenkel L. "Extremely Randomized Trees" *Machine Learning.* 63, 3-42. 2006. doi:10.1007/s10994-006-6226-1
[14] Guo X, Yin Y, Dong C, Yang G, Zhou G. "On the Class Imbalance Problem" *Fourth International Conference on Natural Computation.* 4. 2008. doi:10.1109/ICNC.2008.871
[15] Japkowicz N. "Supervised Versus Unsupervised Binary-Learning by Feedforward Neural Networks" *Machine Learning.* 42, 97-122. 2001. doi:10.1023/A:1007660820062
[16] Jones R et al. "A CDKN2A mutation in familial melanoma that abrogates binding of p16INK4a to CDK4 but not CDK6" *Cancer Res.* 67, 9134-9141. 2007. doi:10.1158/0008-5472.CAN-07-1528
[17] Kiefer J, Wolfowitz J. "Stochastic Estimation of the Maximum of a Regression Function" *Ann. Math. Statis.* 23(3), 462-466. (1952). doi:10.1214/aoms/1177729392
[18] Ng P. "SIFT database" Last visited on April 16 2021. https://sift.bii.a-star.edu.sg/sift4g/
[19] Ng P, Henikoff S. "SIFT: Predicting amino acid changes that affect protein function" *Nucleic Acids Res.* 31(13), 3812-4. 2003. doi:10.1093/nar/gkg509
[20] Pandurangan AP, Blundell TL. "Prediction of impacts of mutations on protein structure and interactions: SDM, a statistical approach, and mCSM, using machine learning." *Protein Sci.* 29, 247-257. 2020. doi:10.1002/pro.3774
[21] Pedregosa F et al. "Scikit-learn: Machine Learning in Python" *Journal of Machine Learning Research.* 12, 2825-2830. 2011.
[22] Ung MU, Lu B, McCammon JA. "E230Q mutation of the catalytic subunit of cAMP-dependent protein kinase affects local structure and the binding of peptide inhibitor" *Biopolymers.* 81, 428-439. 2006. doi:10.1002/bip.20434
[23] Schapire R. "The strength of weak learnability" *Machine Learning.* 5, 197-227. 1990. doi:10.1007/BF00116037
[24] Takamiya O, Seta M, Tanaka K, Ishida F. "Human factor VII deficiency caused by S339C mutation located adjacent to the specificity pocket of the catalytic domain" *Clin. Lab. Haematol.* 24, 233-238. 2002. doi:10.1046/j.1365-2257.2002.00449.x
[25] Wang S et al. "Training deep neural networks on imbalanced data sets" *IEEE.* 4368-4374. 2016. doi:10.1109/IJCNN.2016.7727770
[26] Yamada Y et al. "Catalytic inactivation of human phospholipase D2 by a naturally occurring Gly901Asp mutation" *Arch. Med. Res.* 37, 696-699. 2006. doi:10.1016/j.arcmed.2006.01.006
[27] Yates A et al. "Ensembl 2020" *Nucleic Acids Research.* 48, 682-688. 2019. doi:10.1093/nar/gkz966
[28] Zahra Shamsi, Matthew Chan, Diwakar Shukla. "TLmutation: Predicting the Effects of Mutations Using Transfer Learning" *The Journal of Physical Chemistry B* 124(9), 3845-3854. 2020. doi:10.1021/acs.jpcb.0c00197

A Helping Hand to CDSS Rule Technologies in Fighting Antibiotic Resistance for Hospital Settings

Bernardo Canovas-Segura
AIKE research group (INTICO)
University of Murcia
Murcia, Spain
bernardocs@um.es

Natalia Iglesias
AIKE research group (INTICO)
University of Murcia
Murcia, Spain
natalia.iglesias@um.es

Antonio Morales
AIKE research group (INTICO)
University of Murcia
Murcia, Spain
morales@um.es

Jose M. Juarez
AIKE research group (INTICO)
University of Murcia
Murcia, Spain
jmjuarez@um.es

Manuel Campos
AIKE research group (INTICO)
IMIB-ARRIXACA Institute
University of Murcia
Murcia, Spain
manuelcampos@um.es

***Abstract*—Over the years, an enormous amount of research has been devoted to providing knowledge representation and reasoning on Clinical Decision Support Systems (CDSS). In the last decades, production rules technology has been predominant for CDSS in real hospital settings. Surprisingly, relatively little is understood about the decisions made by engineers to adopt such technology and how new knowledge and rules are updated from existing CDSSs.**

In this work, we put almost 10-years of our research about this topic in perspective, reviewing the work published in [1], [2]. Our efforts have put the spotlight on the antimicrobial resistance problem and how CDSS can help to support antimicrobial stewardship teams in hospitals.

***Index Terms*—Rules, Clinical Decision Support Systems, Clinical Guidelines, Antimicrobial Resistance.**

I. Decision Support in real Hospital Settings

Medical informatics industry is aware of the dynamism of infectious disease research, imposing an urgent need of a quick adaptation of clinical information systems and a robust information exchange. During the last decades, remarkable efforts have been done in developing methodologies and standards for providing fast-upgrade and interoperable ecosystems. However, Clinical Decision Support Systems (CDSS) development still has a monolithic approach consolidating the archipelago model: a set of isolated AI-based tools (islands of knowledge) to support decisions.

There are two key factors to add new knowledge in running CDSSs: the interoperation with other CDSS and new evidence available in the medical literature.

We believe the limited capacity to exchange and update medical knowledge are two sides of the same coin.

To approach the mechanism of knowledge communication between running CDSS is to follow bottom-up strategy, minimising changes in the current representation of information in the CDSSs. From a medical informatics perspective, a common interoperability solution is to use a reference model strategy: to share a standard clinical model, such as SNOMED, to which system involved maps its local terminology.

From the citizen perspective, there is a growing concern on the opacity produced by some AI-systems for human oversight, emphasizing the GDPR-compliant and the need of explanations. According to the European Commission proposal for AI regulation (April 2021), high-risk AI systems, such as those in healthcare, 'should be designed and developed in a way that natural persons can oversee their functioning' [3].

Among the myriad of AI strategies for knowledge representation and reasoning, rule-based approaches are the robust and reliable AI *par excellence*.

In the recent medical literature, some methodologies to extract knowledge are posed [4], [5], while some other efforts are made to interchange standardized knowledge in the form of rules [6]. To date, few studies have investigated methodologies approaching both problems from a holistic perspective in practical scenarios.

In this work, we review our efforts and results during the last decade on the antibiotic resistance medical scenario and the support of Antimicrobial Stewardship Program (ASP) hospital teams. This work summarises such research published in [1], [2] as part of the PhD thesis of Bernardo Canovas-Segura and Natalia Iglesias.

This paper is organized as follows: We first outline the key fundamental ideas underlying our work (section 2), then we review fundamental knowledge sources, addressing the knowledge acquisition issue (section 3). We present our contributions: a framework of criteria for rule technology selection (section 4) and a methodology for extending running CDSS (section 5), in both cases we describe the lesson learned. Finally, we highlight our main conclusions and future works (section 6).

II. General Approach

Ample evidence exists to support clinical decisions using rule-based systems. To date, scant attention has been paid to help CDSS designers who face the problem of updating the knowledge bases from evidence-based medical documents (e.g. clinical guidelines or expert rules). We address this specific issue focusing on two key problems: how to select the most suitable rule technology and, once the technology is adopted, the best strategy to update the knowledge base.

From the scientific point of view, the two main contributions of our efforts were:

- A framework of criteria to select the most suitable rule-base technology for CDSS.
- A methodology to integrate medical knowledge to already running CDSS.

Figure 1 depicts a general overview of our proposal.

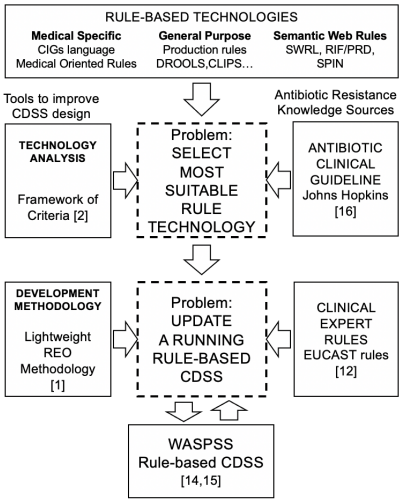


Fig. 1. Key contributions

III. Knowledge Sources and CDSS in Antibiotic Resistance

The progressive increase of antimicrobial resistance and the slow research into new antibiotics have now made the selection of most suitable treatments for infectious diseases more difficult than ever before. For example, despite being one of the most recurrent nosocomial infections in hospitals, the Urinary Tract Infection (UTI) is still a major cause of in-patient deaths throughout the world [7], [8]. Antibiotic Clinical Guidelines are helpful tools for physicians, as they provide optimal antibiotic therapy based on clinical best practices.

A. Antibiotic Clinical Guidelines and CDSS

Clinical Guidelines (CGs) are evidence-based documents that provide useful recommendations about the diagnosis and treatment of diseases. Information encapsulated in CGs is an unquestionable source of knowledge enforce the use of CDSSs.

Information in CGs is based on a systematic review of clinical evidence in a natural language format and, therefore, cannot be easily adopted in CDSSs [9].

Translating CGs into a computable form is costly and it worth when the resulting Computable Guidelines is reusable by different institutions. Unfortunately, CDSSs in real clinical settings cannot deal with complex representations of knowledge and are often limited to simple recommendations based on cause-consequence rule structures.

Such limitations are clearly defined in literature [10], [11], which emphasizes the importance of analysing the feasibility of rule-based formalisms as a realistic way of transforming CGs into computable knowledge for CDSS in real clinical settings.

B. International Expert Rules

Apart from CG, there are other evidence-based sources of medical knowledge. For example, expert knowledge concerning resistance patterns between antimicrobials and bacteria must be included in the CDSS. Such knowledge is often released and updated by transnational healthcare organizations in simple and if-then statement, known as expert rules. This condensed information is helpful for both clinicians and clinical guidelines designers.

The European Committee on Antimicrobial Susceptibility Testing (EUCAST) is an EU organism focused on antimicrobial susceptibility testing, harmonizing the guidelines from EU members [12]. EUCAST periodically releases a set of expert rules to support microbiologists with susceptibility test results. These rules are often used to recommend actions as regards reporting to clinicians. They are based on clinical and/or microbiological evidence, and some rules have an evidence degree, exceptions or comments attached to them. These meta-data play a key role when deciding on a treatment. In general, they are defined mostly over bacteria species (e.g. *IF Streptococcus pneumoniae is resistant to ... THEN ...*) and over families and other groups of bacteria [13].

C. WASPSS: a rule-based CDSS

The Wise Antimicrobial Sterwardship Program Support System (WASPSS) [14], [15]. WASPSS gathers the required information from hospital systems (pharmacy, microbiology, clinical laboratory and the electronic health records) in order to provide decision support in antimicrobial stewardship. To achieve this, it combines business intelligence techniques with a rule-based inference engine to integrate the data and knowledge required in this scenario.

IV. Selection of Rule Technologies

A. Rule technologies

Different criteria is adopted for the design of rules: Derivation rules which obtain new knowledge (implementing sorts of logical implication); Transformation rules checking the truth between statements in different knowledge bases; Integrity constraints on structure to state activity flows; and event-condition-action rules.

Knowledge formalisms are often based on subsets of First Order Logic, finding a trade between expressivity, decidability and efficient inference methods.

The semantics underlying the logics adopted are essential. In Close World Assumption (CWA) something not specifically asserted as true is considered false. In general, Open World Assumption (OWA) adoption can be a problem when all the negative facts have to be asserted.

For medical oriented rules and general-purpose production rules, main implementations are in the form of Rule-Based systems (based on logical Horn clauses) and Semantic Web technology based mainly on Descriptive Logic for ontology representation together with rule extensions.

B. Framework of criteria

The first stage of our research focused on defining a framework of criteria to guide CDSS engineers to select rule-based technologies in order to represent computable CGs in real clinical settings [2].

Firstly, we reviewed existing literature and analysed antibiotic CGs, elaborating an initial list of requirements, which is based on both antibiotic CGs and the most common pitfalls in CDSSs. From this study, we inferred a list of requirements such as: temporal reasoning capacity, event orchestration, universal interoperability, capacity to explain the reasoning process and the compliance with medical standards.

From the above requirements, we identified 3 main categories of analysis for the selection and assessment of rule technologies:

- Language aspects: capacity of the formalism to faithfully contain and represent the knowledge expressed in the original CG's, measured in terms of logic expressivity, user experience, knowledge re-usability, temporal reasoning, explanation of reasoning and knowledge maintenance.
- Interoperability aspects: suitability of the formalism to be easily adopted by different clinical institutions, based on event orchestration, availability of APIs, availability of universal abstract data representations and compliance with medical standards and terminologies.
- Industrial aspects: availability of authoring environments and execution engines, scalability and performance as knowledge evolves, and rich and mature complementary tool ecosystems.

An exhaustive list of Key Performance Indicators (KPI) is provided in [2] to evaluate the capacity of mainstream rule technologies according to these categories of analysis.

C. Use case: Urinary Tract Infection

The John Hopkins Hospital Antibiotic Guidelines (JHH CGs) are an international gold standard and provide recommendations regarding both diagnosis and antibiotic treatment, along with other useful infection management tips [16] that may be helpful in the determination of an Empiric Treatment (ET).

	Criteria	Arden Syntax	Drools	RIF/PRD	OWL+SWRL	OWL+SPIN
LANGUAGE	Logic Expressivity	Horn, CWA, NAF, non-monotonic	Horn, CWA, NAF, non-monotonic	Horn, CWA, no NAF, non-monotonic, Safe vars	DL+Horn-like, No NAF, monotonic, OWA, DL-Safe rules	DL + SPARQL CWA, non monotonic, constraints, own functions and templates
	Explanation of reasoning	Logs	Metadata+logs	Metadata	Reasoners logs	Reasoners logs
	Knowledge maintenance	◐	•	◐	◐	◐
	User experience	◐ (with open source)	•	○	◐	•
	Knowledge sharing	◐Data access (*Curly braces*)	◐Data access POJO Classes	○	◐A-Box data	◐A-Box data
	Temporal reasoning	●	●	○	◐	◐·
INTEROPERABILITY	Event orchestration	● External, Infiniti	● Drools Fusion, jBPM	○	○	◐TopBraidECA-Rule engine
	Integration with third party data stores	● FHIR, JDBC RDBMS, OpenEHR	◐JDBCSQL drivers	○	SPARQL queries, Oracle Plugin	TopBraid FHIR RDF.SPINMAP, DRQ SQL R/O driver
	Compliance with Medical standards	◐FHIR	○	○	◐Taxonomies, vocabularies, W3C	◐Integration of taxonomies, vocabularies, W3C, TopBraid EVN
	Availability of APIs	◐ CDS Hooks API	●Drools KIE API	○	●SWRLAPI, OWLAPI	●OWLAPI, TopBraid SPIN API
INDUSTRY & DEVELOPMENT	Authoring environments (IDEs)	●Arden Suite IDE Medexter ◐Eclipse4Arden IDE (open source)	●Eclipse IDE, Drools Workbench	○	◐SWRLTab Protégé	●TopQuadrant TopBraidComposer
	Execution engines	◐Arden2ByteCode, ●Medexter	●Apache JBOSS ●Eclipse IDE	○	◐ SWRLTab Protégé	●TopQuadrant TopBraidComposer
	Scalability & performance	CPU, Java programs	Memory, algorithm, size of rule base	○	CPU, memory, size of A-box, reasoner, ontology axioms	CPU, size of A-box, reasoner, ontology axioms
	Tool ecosystem	Medexter, Drools, Infiniti, commercial EHRs	Drools Expert, Drools Fusion, Drools Workbench, jBPM	Only proprietary IBM, Oracle	Protégé with Pellet /Hermit reasoner	TopBraid Composer (free edition available), EVN, SPARQL Motion, SPARQL Web Pages GraphQL, SPINx

Legend: ◐ Partially supported/ room for improvement • Full support/well solved ○Not known support/not solved

Fig. 2. Selection criteria of rule technologies for UTI JHH CG [2].

In this use case we studied the suitability of the following rule technologies to represent JHH UTI CG: ARDEN Syntax, DROOLS, RIF/PRD, and the Semantic Web rules technologies (SRWL, SPIN/SHCL). To this end, we took into consideration both a preliminary evaluation using our framework of criteria and a real implementation using such rule technologies.

In order to empirically confirm our framework of criteria we first analysed the JHH UTI CG to infer a list of 81 human readable rules: 56 classification rules, 15 rules event-driven rules triggered by events in the clinical workflow, 6 temporal rules and 4 hybrid rules, which require both temporal reasoning and integration into clinical workflow. With this initial representation and the information available about the rule technologies we obtained the evaluation shown in Figure 2.

We would like to highlight the following issues during the implementation of JHH UTI CG regarding rule technologies:

- ARDEN Syntax: 12 Arden MLMs were coded. Despite their well-known medical orientation, MLMs seems difficult to maintain due to their procedural structure. We remark that the Arden open-source tool lacks robustness, any form of event orchestration or integration using other standards. This confirms the limited score of the language, interoperability and industrial categories of our framework.
- DROOLS: 81 highly atomic rules and 14 agenda groups were coded. It allows to simulate clinical events that trigger rules. Atomicity of knowledge makes its maintenance much simpler, although special care must be taken to keep the right rule execution order, as UTI CGs rules are highly interdependent. Drools supports temporal reasoning and uses metadata and logging to explain the inference process. However, very little is considered about the compliance with medical standards.

- RIF/PRD did not have any workable freely available implementation at the time of our research, so we simply coded our guiding examples to show its syntax, although no test of its correctness was carried out.
- Semantic Web (SW) Rules: We coded a representative subset of the JHH UTI CGs in an OWL 2 ontology - called **UTIO** - with 64 classes, 33 properties and 57 individuals.
  - SWRL: We coded temporal rules in SWRL using the Manchester syntax, allowing temporal reasoning by adding the ontology described in [17] into UTIO.
  - SPIN rules: the TBC Maestro Edition 5.4.0 tool was used, importing UTIO by embedding the SPIN ontology, thus enabling us to use SPIN functions. TBC supports both TTL or the RDF format and the creation of SPIN rules using SPARQL constructs and functions.

  Our findings showed that the OWL reasoner alone could infer 56 of the 81 rules, ranging from diagnosis classification to the determination of the right ET. The required SWRL extension rules had however significant limitations: 1) they were limited to DL-Safe rules, 2) the authoring and execution environment is on the edge of technology, and its evolution has been stopped, 3) as SWRL is monotonic, knowledge cannot be modified or retracted. On the contrary, SPIN and its evolution SHACL (SPIN 2.0) are active developments with a good IDE, supporting incremental reasoning and inference triple materialization

D. Lesson learned

DL is ideal for alerting systems as for terminologies representation, but its reasoning capabilities are limited to the type of DL used, the more expressive the less decidable.

For implementing CGs, Semantic Web languages need extensions in rules for calculations, temporal, and event-driven reasoning. On the contrary, the logics driving SW rules are under the Closed World Assumption(CWA) that we consider closer to human reasoning in clinical settings.

In production rules, atomicity is simple for the maintenance of knowledge. That is, in a rule what you see is what you get, while in an ontology the axioms trigger new knowledge not visible to the naked eye. Debugging issues, rule tracing and the management of high volumes of rules are simple in production rule technologies since they are closer to programming than SW rules.

Finally, data is the fuel of reasoning for both production rules and ontologies. The difference is that in ontologies the data must be incorporated as individuals, while in rules there are usually connectors to databases and reason is directly about them, in the case of Drools using Plain Old Java Objects (POJO) classes.

The lack of standardization in the clinical data of each EHR makes interoperability and knowledge sharing more difficult, both in rules and ontologies.

V. Extending Running CDSSs

The second stage of our research addressed the issue of incorporating new clinical rules for an existing rule based CDSS that is running in a hospital setting. In this study we proposed a straightforward methodology [1] for such purpose, based on the assumptions that changes on the reasoning mechanisms and knowledge representation must be minimised.

The lightweighted methodology adopts the model of reference strategy and implementing REO, a reference ontology. REO represents the minimal set of concepts and relations to update the Knowledge Base with new production rules. This methodology deals with the lack of ontology inference of production rules and the mapping of local terms to a shared knowledge representation.

A. REO methodology

The methodology consists of 4 steps: defining the REO ontology, mapping local terms with REO ontology, extending reasoning properties, and implementing new rules. Figure 3 summarises the REO methodology and we explain these steps in what remains of this section.

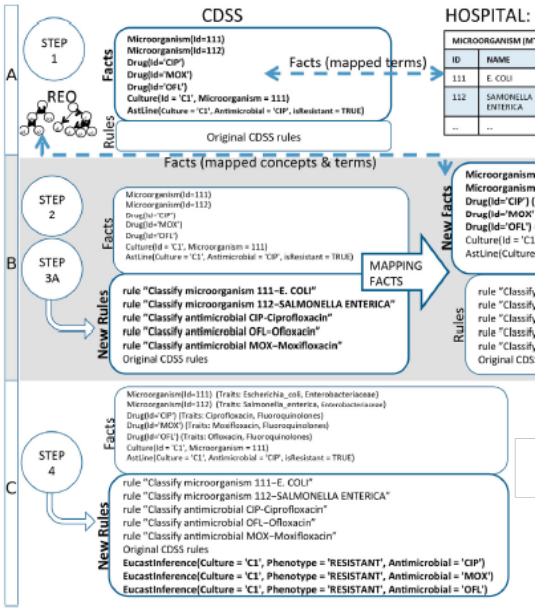


Fig. 3. REO methodology [1].

1) The Reference Ontology: This step aims at designing a reference ontology in accordance with the following criteria:

- Ontology consensus between the clinicians and knowledge engineers.
- Concept granularity considering the definitions of the clinical knowledge of sources and the local terminology used in HIS databases.
- Restrict to a minimal coverage, including only those concepts and relations required by the shared rules.
- Use a subset of standard clinical terminologies when possible, for example ICD, SNOMED, LOINC, ATC, among others.

- T-Box elements only, that is, it must be empty of axioms regarding individuals (concept instances).

2) Local Terminology Mapping: The local concepts must be mapped to the REO ontology as individuals so as to be added to the knowledge based of the CDSS. We consider the following criteria:

- Standard local terminologies: If the local databases use standard clinical terminologies, it might be convenient to incorporate these terminologies into the REO (Step 1) in order to facilitate the mapping process.
- Specific mapping: The terms from the local databases are mapped onto the most specific concept of the REO, that is, the lowest term in the hierarchy.
- A-Box: Mapped terms are asserted, as individuals, in the A-Box of the REO.
- Partial mapping: Not all concepts and terms are mapped. Those concepts of the highest level of abstraction are not mapped onto database elements, but they might be inferred in ontology subsumption reasoning. Database terms might not be mapped onto REO concepts when they are not directly involved in the topic of the clinical knowledge source.

3) Production Rules and Ontology Reasoning: In the third step, the goal is to extend production rule engine with ontology reasoning. We explore two approaches: simulate an ontology reasoning with rules and the use of an external ontology reasoner.

4) Rule Implementation: The last steps aims at implement the rules and integrate them in the CDSS using REO.

B. Use case: Antibiotic Susceptibility

This use case looks at how WASPSS system can incorporate EUCAST expert rules [13]. This experiment was carried out by two engineers and the head of the Infection Control Committee at the collaborating hospital.

1) REO, ATC and NCBI: REO ontology was based on the standard terminologies for organisms and antimicrobials. We used the public organisms taxonomy NCBI [18] that currently contains a total of 552,750 terms. REO for microorganisms consists of 1,714 concepts. For antimicrobials, REO used the available ATC classification [19], containing over 1,900 pharmacological compounds. Finally, 584 concepts were stored in our REO for antimicrobials.

Information coded in the original rules of the WASPSS CDSS belonged to two different hospital systems: microbiology (microorganisms) and pharmacy (antimicrobials). However, both tables are integrated into WASPSS and we, therefore, have access to all of the local terms required from the rules engine. A total number of 1,285 local terms were mapped to REO following a semi-automated procedure.

2) Drools and ontology reasoning: Drools was chosen as the most appropriate RBS engine for WASPSS by following the results of the selection criteria in [2] and the technical requisites of the WASPSS project. The third step of REO methodology aims to extend Drools with ontology reasoning properties. We implemented two different approaches. From the one hand, ontology subsumption inference was simulated using the *traits* capability from Drools, which allows dynamic multi-hierarchical typing, and alternatively, using RuQAR [20] that translates ontology triples to Drools facts. On the other hand, an external ontology reasoner, HermiT [21], was integrated in the system and queried during rule evaluation when concept subsumption was required.

3) EUCAST rules implemented: The implementation process for coding EUCAST expert rules took approximately two weeks and another week for the verification process. A total of 90 Drools rules were coded including intrinsic/interpretive resistance and exceptional phenotypes rules.

C. Lesson learned

Our experiments provided convincing evidence of the suitability of our simple methodology to combine ontology and rule-based reasoning for interoperability between CDSS. These findings are less surprising if we consider knowledge source is expert rules, which it clearly eases the knowledge acquisition step. From a technical point of view, the use of an external ontology reasoner obtained better performance in our experiments as regards both the initialization and execution times, although this approach may not be possible to implement in other rule-based engines. On the other hand, the usage of multi-hierarchical typing or to include the subsumption relationships as facts are strategies easier to apply to other implementations.

Our findings are not generalizable beyond antibiotic support scenarios and expert rules, however in our experience, the use of interpretable structures, such as archetypes, used to overcome the lack of data standards in clinical domains. In this sense, in [22], the authors suggest the use of openEHR and archetypes to intercommunicate medical records and CDSSs.

Despite the advances made as regards the acceptance of CDSS in daily practice, little efforts have recently been made to develop *methodologies* with which to extract and integrate knowledge for CDSS. We believe that the REO methodology is a clear contribution in this direction.

VI. Conclusions and challenges

Our experience is limited to the clinical problems of infectious diseases and antimicrobial stewardship. Nonetheless, they are practical examples that we believe might be useful in other clinical scenarios.

From a practical point of view, the selection of the appropriate rule technology is not a trivial step, and many facets should be evaluated. We consider that our work in [1] can be helpful to bring some guidance in the myriad of options.

Once a technology is chosen, to follow a methodology can be helpful to avoid common pitfalls that might lead to systems that become difficult to update or to incorporate into an interconnected ecosystem of HISs. The REO methodology followed in [2] is a successful case of use that can be used and adapted when updating or even designing future CDSSs.

Expressivity and flexibility provided by SW rules, to some extent overcomes the reasoning needs of rule-based CG implementations offering superb classification and subsumption, ideal for terminologies or categories.

Production rules are solid but might require strong background on programming, making more difficult the knowledge elicitation when knowledge sources requirement become complex (e.g. guidelines). However, in our analysis, few CG recommendations require a high demanding reasoning capacity. Therefore, according to our study, we support the use of production rules technology combined, if needed, with some punctual support of SW reasoning engines.

According to our experiments, all the rule technologies examined must address design considerations about how to interoperate with external Health Information Systems (HIS). The lack of data standardization, the peculiarities of rule languages and the availability of robust tools and ecosystems limit the computable representation of knowledge and its effective sharing between institutions. The interplay between data, knowledge and processes must be as clean as possible, with clear interfaces and loose coupling. Atomic knowledge is easier to maintain by knowledge experts, than if intertwined with institution-specific processes or data. Universal data is therefore a prerequisite for interoperability and knowledge sharing. In that sense new initiatives such as openEHR can offer an outcome, as they use re-usable shareable archetypes for clinical data-concepts.

Rule technologies present excellent capabilities for explaining and tracing their outputs and, therefore, they have been a relevant part of most CDSSs. Furthermore, the usage of these kind of technologies will be surely boosted in the future given the increasing concern about explainable AI in healthcare.

Acknowledgement

This work was partially funded by the Spanish Ministry of Science, Innovation and Universities (MCIU), the Spanish Agency for Research (AEI), the European Fund for Regional Development (EFRD) under the SITSUS project (Ref: RTI2018-094832-B-I00), and by the National Network IABIOMED_NET (Ref: RED2018-102312-T).

References

[1] B. Cánovas-Segura, A. Morales, J. M. Juarez, M. Campos, and F. Palacios, "A lightweight acquisition of expert rules for interoperable clinical decision support systems," *Knowledge-Based Systems*, vol. 167, pp. 98–113, 2019. [Online]. Available: http://hdl.handle.net/10201/107023

[2] N. Iglesias, J. M. Juarez, and M. Campos, "Comprehensive analysis of rule formalisms to represent clinical guidelines: Selection criteria and case study on antibiotic clinical guidelines," *Artificial Intelligence in Medicine*, vol. 103, p. 101741, 2020.

[3] COM(2021) 206 final, "Laying down harmonised rules on artificial intelligence (artificial intelligence act) and amending certain union legislative acts," European Commission, Brussels, Tech. Rep., 2021. [Online]. Available: http://www.whocc.no/filearchive/publications/1_2013guidelines.pdf

[4] Y.-F. Zhang, Y. Tian, T.-S. Zhou, K. Araki, and J.-S. Li, "Integrating hl7 rim and ontology for unified knowledge and data representation in clinical decision support systems," *Computer Methods and Programs in Biomedicine*, vol. 123, no. C, pp. 94–108, Jan. 2016. [Online]. Available: http://dx.doi.org/10.1016/j.cmpb.2015.09.020

[5] S. Zolhavarieh and D. Parry, "KQA: A knowledge quality assessment model for clinical decision support systems," in *MEDINFO 2017: Precision Healthcare through Informatics - Proceedings of the 16th World Congress on Medical and Health Informatics, Hangzhou, China, 21-25 August 2017.*, 2017, pp. 983–986. [Online]. Available: https://doi.org/10.3233/978-1-61499-830-3-983

[6] M. Samwald, K. Fehre, J. de Bruin, and K.-P. Adlassnig, "The Arden Syntax standard for clinical decision support: Experiences and directions," *Journal of Biomedical Informatics*, vol. 45, no. 4, pp. 711–718, 2012.

[7] A. Flores-Mireles, J. Walker, M. Caparon, and S. Hultgren, "Urinary tract infections: Epidemiology, mechanisms of infection and treatment options," *Nature Reviews Microbiology*, vol. 13, 04 2015.

[8] T. Rawson, L. Moore, B. Hernandez, E. Charani, E. Castro-Sánchez, P. Herrero, B. Hayhoe, W. Hope, P. Georgiou, and A. Holmes, "A systematic review of clinical decision support systems for antimicrobial management: Are we failing to investigate these interventions appropriately?" *Clinical Microbiology and Infection*, vol. 23, 03 2017.

[9] P. Romano and A. Splendiani, "Applications of semantic web methodologies and techniques to biology and bioinformatics," 01 2008, pp. 200–239.

[10] D. F. Sittig, A. Wright, J. A. Osheroff, B. Middleton, J. M. Teich, J. S. Ash, E. Campbell, and D. W. Bates, "Grand challenges in clinical decision support," *Journal of Biomedical Informatics*, vol. 41, no. 2, pp. 387–392, 2008.

[11] K. Kawamoto, G. Del Fiol, D. Lobach, and R. Jenders, "Standards for scalable clinical decision support: Need, current and emerging standards, gaps, and proposal for progress," *The open medical informatics journal*, vol. 4, pp. 235–44, 12 2010.

[12] "EUCAST - European Committee on Antimicrobial Susceptibility Testing," http://www.eucast.org/. [Online]. Available: http://www.eucast.org/

[13] R. Leclercq, R. Cantón, D. F. J. Brown, C. G. Giske, P. Heisig, a. P. Macgowan, J. W. Mouton, P. Nordmann, a. C. Rodloff, G. M. Rossolini, C. J. Soussy, M. Steinbakk, T. G. Winstanley, and G. Kahlmeter, "EUCAST expert rules in antimicrobial susceptibility testing," *Clinical Microbiology and Infection*, vol. 19, no. 2, pp. 141–160, 2013.

[14] F. Palacios, M. Campos, J. M. Juarez, S. E. Cosgrove, E. Avdic, B. Cánovas-Segura, A. Morales, M. E. Martínez-Nuñez, T. Molina-García, P. García-Hierro, and J. Cacho-Calvo, "A clinical decision support system for an Antimicrobial Stewardship Program," in *HEALTHINF 2016 - 9th International Conference on Health Informatics, Proceedings*. Rome: SciTePress, 2016, pp. 496–501.

[15] B. Cánovas-Segura, M. Campos, A. M. Nicolas, J. M. Juarez, and F. Palacios, "Development of a clinical decision support system for antibiotic management in a hospital environment," *Progress in Artificial Intelligence*, vol. 5, no. 3, pp. 181–197, 2016.

[16] The Johns Hopkins Hospital Antimicrobial Stewardship Program, "Jhh guidelines for antibiotic use," The Johns Hopkins Hospital, Tech. Rep., 2018. [Online]. Available: https://www.hopkinsmedicine.org/antimicrobial-stewardship/guidelines/

[17] M. J. O'Connor and A. K. Das, "A method for representing and querying temporal information in owl," in *Biomedical Engineering Systems and Technologies*, A. Fred, J. Filipe, and H. Gamboa, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 97–110.

[18] S. Federhen, "The NCBI Taxonomy database," *Nucleic acids research*, vol. 40, no. Database issue, pp. D136–D143, jan 2012.

[19] WHO Collaborating Centre for Drug Statistics Methodology, "Guidelines for ATC classification and DDD assignment 2013," WHO Collaborating Centre for Drug Statistics Methodology, Oslo, Tech. Rep., 2012. [Online]. Available: http://www.whocc.no/filearchive/publications/1_2013guidelines.pdf

[20] J. Bak, M. Nowak, and C. Jedrzejek, "Ruqar: Reasoning framework for owl 2 rl ontologies," in *The Semantic Web: ESWC 2014 Satellite Events*, V. Presutti, E. Blomqvist, R. Troncy, H. Sack, I. Papadakis, and A. Tordai, Eds. Cham: Springer International Publishing, 2014, pp. 195–198.

[21] R. Shearer, B. Motik, and I. Horrocks, "HermiT : A Highly-Efficient OWL Reasoner," *Complexity*, vol. 432, p. 10, 2008.

[22] M. Marcos, J. A. Maldonado, B. Martínez-Salvador, D. Boscá, and M. Robles, "Interoperability of clinical decision-support systems and electronic health records using archetypes: A case study in clinical trial eligibility," *Journal of Biomedical Informatics*, vol. 46, no. 4, pp. 676–689, 2013.

Aprendizaje automático para la detección temprana de la enfermedad de Parkinson

Maitane Martinez-Eguiluz
Olatz Arbelaitz
Ibai Gurrutxaga
Javier Muguerza
and Iñigo Perona
Dept. of Computer Architecture and Technology,
University of the Basque Country (UPV/EHU), Donostia, Spain
mmartinez149@ikasle.ehu.eus
http://www.aldapa.eus

Ane Murueta-Goyena
Dept. of Preventive Medicine and Public Health
University of the Basque Country (UPV/EHU), Leioa, Spain
Biocruces Bizkaia Health Research Institute,
Barakaldo, Spain
ane.muruetagoyena@ehu.eus

Iñigo Gabilondo
Biocruces Bizkaia Health Research Institute,
Barakaldo, Spain
IKERBASQUE,
Basque Foundation for Science,
Bilbao, Spain
inigo.gabilondocuellar@osakidetza.eus

***Resumen*—La enfermedad de Parkinson (EP) es el segundo trastorno neurodegenerativo más común del mundo. Por lo tanto, es importante detectarla de forma temprana. Los síntomas característicos de la EP son las alteraciones motoras, como el temblor involuntario en reposo, los movimiento lentos, rigidez muscular o problemas de equilibrio. A día de hoy, el diagnóstico de EP se basa en estos síntomas cardinales. Sin embargo, los pacientes con EP desarrollan manifestaciones no motoras que preceden a los síntomas motores hasta 20 años. Entre los síntomas no motores más comunes podemos encontrar la anosmia o olfacción alterada, trastorno del sueño REM, estreñimiento, depresión, trastornos cognitivos y otros. Por ello, existe un creciente interés en explorar la utilidad de los síntomas no motores para el diagnóstico temprano de la EP. Este trabajo pretende contribuir al diagnóstico precoz de la EP basándose en datos no motores de 490 pacientes con EP Idiopática y 197 sujetos de control. Para ello, se ha utilizado una base de datos de un repositorio internacional (PPMI), del cual se han seleccionado variables no motoras. Se han generado 4 versiones de bases de datos con granularidad creciente que se emplearán para entrenar 5 clasificadores con algoritmos de aprendizaje automático (Naive Bayes, Support Vector Machine, Multi-Layer Perceptron, K-Nearest Neighbors y Random Forest). A través de estos algoritmos se detectó un conjunto de variables que permitían diferenciar pacientes de controles, lo que sugiere que se podría realizar una detección temprana de la EP utilizando una versión reducida de las pruebas y los cuestionarios.**

I. Introduction

La enfermedad de Parkinson (EP) es un trastorno neurodegenerativo de evolución lenta que se caracteriza por la pérdida de neuronas dopaminérgicas (productoras de la dopamina) en la sustancia negra y otras estructuras cerebrales [1], [2], [3]. La enfermedad fue descrita por primera vez en 1817 por James Parkinson [4]. Esta enfermedad crónica es el segundo trastorno neurodegenerativo más común, después del Alzheimer [5], y su prevalencia va en aumento: de 1990 a 2015 el número de pacientes con EP se ha duplicado en el mundo [6] y actualmente afecta al 1 % de las personas mayores de 60 años [7]. En España, hay entre 80.000 y 100.000 pacientes de EP y cada año se diagnostican 8.000 nuevos casos.

Hoy en día, la enfermedad no tiene cura, y los tratamientos son meramente sintomáticos para mejorar la calidad de vida de los pacientes. El diagnóstico precoz permitiría tratar la enfermedad con agentes neuroprotectores, actualmente en ensayos clínicos, y posibilitarían frenar la progresión de la pérdida neuronal.

La pérdida de dopamina provoca síntomas motores en los pacientes con EP, como un temblor involuntario en reposo, movimientos lentos (bradicinesia), problemas de equilibrio y rigidez [3], [5]. Estos síntomas cardinales aparecen cuando el 50 %-70 % de las neuronas dopaminérgicas en sustancia negra han degenerado (llamada fase motora [8]). Además, los pacientes con EP desarrollan manifestaciones no motoras, como trastornos del estado de ánimo y del sueño, pérdida del olfato y disfunción del sistema nervioso autónomo y trastornos cognitivos [7], [5]. Estos síntomas aparecen en las primeras fases de la enfermedad [3] (llamada fase prodrómica [8]).

La enfermedad de Parkinson Idiopática (EPI) es la etiología más común, representando el 60-75 % de los casos [9].

Hoy en día, el diagnóstico de la EP se basa en la sintomatología clínica, lo cual dificulta un diagnóstico certero, considerando que tanto los síntomas que presentan como la gravedad de los mismos varía de un paciente a otro. Además, la evolución clínica no es común para todos los pacientes y, aunque existan ciertos indicios que apunten a una peor evolución, el pronóstico en el momento del diagnóstico es incierto.

En este contexto, es importante identificar biomarcadores para la detección temprana de la EP. Los biomarcadores son características físicas que pueden medirse objetivamente y están relacionadas con el estado y la progresión de una enfermedad. Aunque en la EP los síntomas más destacados son los motores, los síntomas no motores normalmente preceden

a los motores [7]. Por esta razón, este trabajo tiene como objetivo proporcionar un enfoque para la detección temprana de la EP utilizando características no motoras para diferenciar entre la EPI y los controles sanos (CS). Si tenemos en cuenta que, por un lado, estas características son fáciles de obtener —ya que se obtienen mediante la realización de una serie de pruebas y cuestionarios—, y que, por otro lado, los síntomas no motores se inician en la primera fase de la enfermedad, lo que proponemos en este trabajo es una primera aproximación a un sistema de detección precoz de la EP fácilmente implementable en atención primaria, con el fin de minimizar el número de pruebas que se realizan al paciente, para agilizar la primera evaluación de la enfermedad, y detectarla en fases tempranas de su desarrollo.

El documento procede a resumir algunos trabajos relacionados en la Sección II y continúa describiendo el tipo de datos y pruebas utilizadas en este trabajo en la Sección III. Las características de las bases de datos generadas para los experimentos se describen en la Sección IV y los métodos de aprendizaje automático y las métricas utilizadas para los experimentos se describen en la Sección V. En la Sección VI describimos los experimentos realizados y los resultados, para terminar con unas observaciones finales en la Sección VII.

II. Contexto

El uso de las técnicas de minería de datos se ha incrementado en varias áreas en las últimas dos décadas [10]. En el ámbito médico, concretamente, la Organización Mundial de la Salud (OMS) identificó en 1997 el potencial del uso de estas técnicas para mejorar el diagnóstico y la predicción médica mediante el uso de repositorios de datos médicos [11].

Los trabajos que utilizan técnicas de aprendizaje automático son numerosos y variados y pueden clasificarse según el tipo de tarea, la naturaleza de los datos utilizados, las técnicas o las bases de datos. Por ejemplo, si nos centramos en el tipo de tarea, hay muchos proyectos cuyo objetivo principal es diferenciar a los pacientes con EP de los sujetos de CS.

Muchos de ellos se basan en exámenes neurológicos basados en imágenes. Algunos investigadores utilizan imágenes de resonancia magnética (RM) [12], [6], mientras que otros utilizan imágenes DaT-SCAN [2], [13], [14], que permite visualizar las terminaciones nerviosas dopaminérgicas en el cuerpo estriado. Otros estudios combinan datos de imágenes con datos clínicos [15] o como Rahmim et al. [1], quienes combinan las características de imágenes RM y DaT-SCAN con otras características clínicas para predecir la gravedad motora en pacientes con EP en cuatro años. Una de las formas más comunes de medir la gravedad de estos síntomas es utilizando la tercera parte del test MDS-UPDRS[1] (prueba médica para la evaluación del estado motor). Por ejemplo, Cavallo et al. [16] adquieren datos sobre el movimiento de las extremidades superiores.

Otra forma de realizar la tarea de clasificación binaria es usando señales de voz [3], [17], [18], [19].

[1]https://www.movementdisorders.org/MDS/MDS-Rating-Scales/MDS-Unified-Parkinsons-Disease-Rating-Scale-MDS-UPDRS.htm

Las líneas anteriores demuestran que las técnicas de aprendizaje automático se están convirtiendo en algo habitual en el campo de la detección y el diagnóstico de la EP, sin embargo, se pueden encontrar pocos trabajos centrados en la detección precoz utilizando únicamente síntomas no motores.

III. Datos

Los datos utilizados en este trabajo son items no motores obtenidos de la base de datos Parkinson's Progression Markers Initiative (PPMI).

III-A. PPMI database

PPMI es un estudio que identifica biomarcadores de la progresión de la EP para mejorar la investigación terapéutica y etiológica [20]. El estudio se realiza a través de una asociación público-privada financiada por la Fundación Michael J. Fox para la Investigación del Parkinson. En el marco de este estudio se han adquirido datos clínicos, incluyendo cuestionarios y escalas neurológicas, muestras biológicas y pruebas de imagen. Para la investigación se reclutaron dos tipos de sujetos procedentes de 24 centros de estudio: pacientes con EP y sujetos CS de edad y sexo similares. En el momento de la inscripción, los sujetos con EP debían tener al menos 30 años, estar diagnosticados en los dos últimos años y no haber recibido tratamiento farmacológico para la enfermedad [21]. Además, estos sujetos se sometieron a la toma de imágenes del transportador de dopamina para corroborar la disminución en los transportadores de dopamina presinápticos, requisito utilizado para el diagnóstico diferencial de EP. En cuanto a los sujetos CS, en el momento de la inscripción requerían una edad de 30 años o más sin un trastorno neurológico activo.

III-B. Cuestionarios y pruebas de síntomas no motores

Existen varios tipos de cuestionarios y pruebas que se centran en la detección de uno o varios síntomas no motores. Los cuestionarios constan de preguntas escritas a las que el sujeto debe responder eligiendo alguna de las opciones ofrecidas. Las pruebas clínicas, sin embargo, son pruebas o ejercicios diseñados para evaluar conocimientos, habilidades, etc. Para este trabajo se han seleccionado los siguientes cuestionarios y pruebas clínicas.

- Epworth Sleepiness Scale (ESS) [22] es un cuestionario con 8 preguntas, en el que el sujeto califica las posibilidades habituales de quedarse dormido mientras realiza 8 actividades cotidianas. A mayor puntuación, más propensa es la persona a padecer somnolencia diurna.
- Geriatric Depression Scale (GDS) [23] es un cuestionario que representa una escala de detección de depresión con 15 preguntas de sí o no.
- SCOPA-AUT [24] es un cuestionario sobre los problemas que se han producido en diversas funciones corporales (concretamente, urinarias, gastrointestinales, cardiovasculares, termorreguladoras, pupilomotoras y sexuales) durante el último mes.
- Symbol Digit Modalities Test (SDMT) [25] es una prueba neuropsicológica que mide la atención y velocidad de

procesamiento, principalmente visual. Esta prueba consiste en emparejar, en un tiempo limitado, 120 símbolos con sus correspondientes números del 1 al 9.

- Benton Judgment of Line Orientation Test (BJLOT) [26] es una prueba neuropsicológica que evalúa el dominio visoespacial, mediante la tarea de discriminar correctamente la dirección de las líneas.
- University of Pennsylvania Smell Identification Test (UPSIT) [27] es una prueba para evaluar el funcionamiento del sistema olfativo de una persona.
- Montreal Cognitive Assessment (MoCA) [28] es una prueba de cribado utilizada para detectar el deterioro cognitivo.
- Hopkins Verbal Learning Test - Revised (HVLTR) [29] es una prueba que evalua el aprendizaje y la memoria verbal.

IV. Bases de datos

Extrajimos del PPMI un conjunto de 687 sujetos: 490 de ellos (71 %) con EPI y 197 (29 %) eran CS. El 34 % eran mujeres y el 66 % restante eran hombres. Los atributos utilizados en este trabajo se dividen en tres grupos: demográficos y clínicos, resultados en cuestionarios y resultados de las pruebas clínicas. Se seleccionaron los resultados obtenidos en la visita inicial de cada sujeto, creando una base de datos *baseline* donde los sujetos con EP llevaban conviviendo con la enfermedad un máximo de 3 años.

Generamos 2 versiones de la base de datos: una primera en la que todas las preguntas o pruebas se consideran individualmente (base de datos individual, BDI) y una segunda en la que se crea una única variable para cada prueba o cuestionario como resultado total (base de datos total, BDT). Los atributos generales y clínicos incluyen: género (GENDER), años de educación (EDUCYRS), mano dominante (HANDED), edad (AGE) y la propia clase (EPI o CS). Los cuestionarios y las pruebas son los descritos en la Sección III-B. La Tabla I resume las variables de las dos versiones de la base de datos. El BDI tiene 106 variables mientras que el BDI tiene 17 variables, siendo siete de ellas calculadas acumulando la respuesta a las preguntas o pruebas en el mismo cuestionario. La agrupación o inclusión de variables individuales en la BDT se ha realizado bajo la supervisión de expertos médicos.

Individual, BDI	**Total, BDT**
GENDER	GENDER
EDUCYRS	EDUCYRS
HANDED	HANDED
AGE	AGE
ESS1, ESS2, ESS3, ESS4, ESS5, ESS6, ESS7, ESS8	ESS_total
GDSSATIS, GDSBORED, GDSGSPIR, GDSHLPLS, GDSWRTLS, GDSHOPLS, GDSDROPD, GDSEMPTY, GDSAFRAD, GDSHAPPY, GDSHOME, GDSMEMRY, GDSALIVE, GDSENRGY, GDSBETER	GDS_total
SCAU1, SCAU2, SCAU3, SCAU4, SCAU5, SCAU6, SCAU7, SCAU8, SCAU9, SCAU10, SCAU11, SCAU12, SCAU13, SCAU14, SCAU15, SCAU16, SCAU17, SCAU18, SCAU19, SCAU20, SCAU21, SCAUSEX1, SCAUSEX2, SCAU23A	SCAU_total
SCAU26A	SCAU26A
SCAU26B	SCAU26B
SCAU26C	SCAU26C
PTCGBOTH	–
SDMTOTAL	SDMTOTAL
BJLOTPAR1, BJLOTPAR2, BJLOTPAR3, BJLOTPAR4, BJLOTPAR5, BJLOTPAR6, BJLOTPAR7, BJLOTPAR8, BJLOTPAR9, BJLOTPAR10, BJLOTPAR11, BJLOTPAR12, BJLOTPAR13, BJLOTPAR14, BJLOTPAR15	BJLOT_total
UPSITBK1, UPSITBK2, UPSITBK3, UPSITBK4	UPSIT_total
MCAALTTM, MCACUBE, MCACLCKC, MCACLCKN, MCACLCKH, MCALION, MCARHINO, MCACAMEL, MCAFDS, MCABDS, MCAVIGIL, MCASER7, MCASNTNC, MCAVF, MCAABSTR, MCAREC1, MCAREC2, MCAREC3, MCAREC4, MCAREC5, MCADATE, MCAMONTH, MCAYR, MCADAY, MCAPLACE, MCACITY	MCATOT
MCAVFNUM	–
HVLTRT1, HVLTRT2, HVLTRT3	HVLTRT_total
HVLTRDLY	HVLTRDLY
HVLTREC	HVLTREC
Class	Class

Cuadro I: Resumen de las variables utilizadas en cada una de las bases de datos

V. Métodos

Como primera aproximación, proponemos comparar varios clasificadores construidos directamente sobre las dos bases de datos descritas en la sección anterior y los mismos clasificadores construidos sobre las bases de datos tras un proceso de selección de características. El uso del método de selección de características se debe a dos razones. Por un lado, uno de los objetivos de este trabajo es identificar las variables más significativas. Por otro lado, estas técnicas ayudan a simplificar los modelos y facilitan la adquisición de información y su interpretación, además de reducir los tiempos de entrenamiento y las probabilidades de sobreajuste. Entre las estrategias de selección de características, utilizamos los métodos de filtrado porque tienen menor coste computacional y no están ajustados a un tipo específico de modelo de predicción [30], y, en consecuencia, pueden combinarse con diferentes clasificadores. Utilizamos el método multivariante denominado Selección de Características Correlacionadas (SCC) porque tratar con características redundantes, duplicadas y correlacionadas.

El método SCC fue desarrollado por Hall y Smith [31] y busca subconjuntos correlacionados con la clase pero independientes entre sí. El algoritmo asume que las características irrelevantes tienen una baja correlación con la clase, por lo que no tienen que ser incluidas en los subconjuntos. Además, examinan los atributos excesivos, ya que suelen estar correlacionados con alguno de los otros atributos. En este proyecto se aplica un algoritmo voraz como heurístico de búsqueda.

Para evaluar el subconjunto S de k características, se utiliza la siguiente fórmula:

$$Merit_s = \frac{k \cdot \overline{r_{cf}}}{\sqrt{k + k(k-1)\overline{r_{ff}}}} \quad (1)$$

donde $\overline{r_{cf}}$ es el valor medio de correlación entre la clase y las características, y $\overline{r_{ff}}$ es el valor medio de correlación

entre todos los pares de características.

Para implementar el filtro SCC se ha usado el software Weka. Todo lo restante, es decir, el análisis de datos, la implementación de los algoritmos, etc. se ha implementado usando la librería scikit de Python.

V-A. Clasificadores

Los experimentos se realizaron con cinco clasificadores de aprendizaje automático, cuyos parámetros eran los de defecto. Se describen a continuación:

- **Naive Bayes (NB)** [32] es un clasificador probabilístico basado en la aplicación del teorema de Bayes [33]. Se llama *naive* porque asume que una característica es independiente de cualquier otra característica dada la variable de clase. Utilizamos el algoritmo Gaussian Naive Bayes [34], que asume que la probabilidad de las características es gaussiana.
- **Support Vector Machine (SVM)** son un conjunto de algoritmos de aprendizaje supervisado que construyen un hiperplano o conjunto de hiperplanos en un espacio de muy alta dimensión para separar las clases. Para resolver problemas no lineales, los datos se transforman mediante kernels [35], en nuestro caso hemos utilizado el kernel de la Función de Base Radial (FBR).
- **Multi-Layer Perceptron (MLP)** [36] es una red neuronal artificial que consta de 3 tipos de capas (capa de entrada, capa oculta y capa de salida) vinculadas entre sí y compuestas de un conjunto de nodos. Excepto los nodos de la capa de entrada, cada nodo es una neurona que utiliza una función de activación no lineal [37], ReLU en nuestro caso. Esta red se entrena mediante la técnica de *ADAM* [38] y puede distinguir los datos que no son linealmente separables [39]. Implementamos tres arquitecturas con 50, 100 y 200 neuronas en la capa oculta y las entrenamos durante 150 épocas.
- **K-Nearest Neighbors (KNN)** [40] es un método de clasificación no paramétrico, es decir, no asume ninguna distribución sobre los datos subyacentes. El algoritmo calcula la distancia (distancia euclidiana en este caso) de una nueva observación a las observaciones de entrenamiento, selecciona las K observaciones más cercanas y asigna la nueva observación a la clase a la que pertenecen la mayoría de las K observaciones (3 o 5 en este trabajo).
- **Random Forest (RF)** [41] construye un conjunto de árboles de decisión [42] (100 en este caso) de forma paralela, cada uno entrenado en una muestra ligeramente diferente generada por *bootstrapping* [43] y limitando aleatoriamente las características que se utilizarán en cada árbol. Para predecir una nueva observación, todos los árboles de decisión se utilizan de forma independiente y la nueva observación se asigna a la clase más común.

VI. Experimentos y resultados

Se utilizaron cuatro bases de datos para los experimentos. Las dos originales, BDI con 106 características y BDT con 17 características, y las dos versiones de bases de datos obtenidas tras aplicar SCC a ambas para seleccionar características. El SCC redujo considerablemente el conjunto de características, concretamente a 13 características (BDI+SCC) y 4 características (BDT+SCC), como se indica a continuación:

- BDI+SCC: AGE, GDSENRGY, SCAU2, SCAU6, SCAU9, SCAU26B, UPSITBK1, UPSITBK2, UPSITBK3, UPSITBK4, MCASER7, MCAVFNUM, MCAREC2
- BDT+SCC: AGE, SCAU_total, UPSIT_total, MCATOT

A partir de estas listas podemos observar que la importancia de las variables individuales suele coincidir con la importancia de las variables acumuladas, pero con algunas diferencias. Mientras que la edad, SCOPA-AUT, UPSIT y MoCA parecen ser importantes en ambas opciones, la información de GDS desaparece en la segunda. Ninguna de las dos opciones propone el uso de las pruebas ESS, SDMT, BJLOT y HVLTR.

Se entrenaron y probaron ocho clasificadores (NB, SVM, RF, MLP50, MLP100, MLP200, KNN3 y KNN5) con las cuatro versiones de bases de datos. Se utilizó la codificación *one-hot encoding* con características nominales y se utilizó la normalización min-max para reescalar las variables continuas.

Para evitar el sobreajuste, los modelos se validaron mediante la validación cruzada estratificada en 10 iteraciones (10-fold-CV) [44]. En concreto, se realizaron 5 ejecuciones de 10-fold-CV. En cuanto a las métricas de rendimiento utilizadas para evaluar las diferentes soluciones, la precisión (Acc) y el Valor-F (F1) [45] como medida de la fiabilidad del modelo, calculada a partir de la media armónica entre Precisión y Exactitud.

Las Tablas II y III resumen los resultados obtenidos para cada clasificador y versión de la base de datos. Los mejores resultados de cada base de datos aparecen marcados en negrita, y la celda correspondiente a los mejores resultados entre todos aparece sombreada.

El primer punto a destacar es que todos los clasificadores obtienen resultados aceptables, lo que demuestra que los síntomas no motores pueden utilizarse para la detección temprana de la EP. En cuanto a cómo afecta la selección de características a los resultados, en la Tabla II muestra que en el caso de BDI + SCC, en todos los clasificadores menos en RF, además de construirse modelos más simples, la calidad de los resultados mejora. La mejora es considerable para los clasificadores con peor rendimiento con todo el conjunto de características: NB y KNN; y menor, en el caso de los clasificadores con mejor rendimiento con las 106 características: SVM y MLP. No obstante, en todos los casos el número de variables se reduce considerablemente (13 características en lugar de utilizar 106). Para la BDT, al ser una base de datos mucho más sencilla, la mejora obtenida con la aplicación de SCC es en general menor. Sólo se mantiene en el caso de NB, SVM, MLP200, KNN3 y KNN5.

Ninguno de los clasificadores ha conseguido los mejores resultados utilizando la base de datos de la BDI con 106 características, y ésta suele ser la que obtiene los peores resultados. Por tanto, los resultados muestran que la agrupación de elementos o la selección de los mismos no solo simplifica los sistemas, sino que también mejora el rendimiento de los

	Metrics		NB	SVM	RF	MLP50	MLP100	MLP200	KNN3	KNN5
BDI	Acc	(MA)	0.569	0.821	**0.834**	0.824	0.832	0.834	0.701	0.729
		(DE)	0.054	0.040	0.043	0.042	0.044	0.038	0.057	0.063
	F1	(MA)	0.588	0.879	**0.887**	0.879	0.883	0.886	0.790	0.811
		(DE)	0.069	0.027	0.030	0.030	0.031	0.026	0.044	0.048
BDI+SCC	Acc	(MA)	0.815	**0.855**	0.827	0.826	0.835	0.843	0.823	0.828
		(DE)	0.044	0.047	0.039	0.047	0.046	0.048	0.035	0.041
	F1	(MA)	0.862	**0.900**	0.880	0.878	0.885	0.891	0.874	0.876
		(DE)	0.034	0.032	0.027	0.033	0.034	0.034	0.026	0.030

Cuadro II: Rendimiento de los clasificadores con la versión individual. MA = Media Aritmética y DE = Desviación Estándar

	Metrics		NB	SVM	RF	MLP50	MLP100	MLP200	KNN3	KNN5
BDT	Acc	(MA)	0.764	0.820	**0.849**	0.836	0.838	0.839	0.757	0.760
		(DE)	0.054	0.030	0.043	0.044	0.045	0.050	0.052	0.044
	F1	(MA)	0.824	0.879	**0.897**	0.889	0.887	0.889	0.831	0.836
		(DE)	0.043	0.020	0.029	0.030	0.032	0.036	0.038	0.032
BDT+SCC	Acc	(MA)	0.799	**0.847**	**0.847**	0.824	0.833	0.840	0.837	0.824
		(DE)	0.054	0.046	0.048	0.045	0.046	0.041	0.042	0.046
	F1	(MA)	0.847	0.893	**0.894**	0.879	0.886	0.888	0.885	0.875
		(DE)	0.043	0.033	0.034	0.032	0.033	0.030	0.030	0.034

Cuadro III: Rendimiento de los clasificadores con la versión total. MA = Media Aritmética y DE = Desviación Estándar

clasificadores. Los mejores resultados se obtuvieron con el clasificador SVM construido a partir de la base de datos BDI + SCC, lo que significa que la clasificación pudo realizarse utilizando sólo 13 características, 12 ítems no motores, incluyendo preguntas individuales de distintos cuestionarios y secciones de diferentes pruebas, y la edad.

VI-A. Debate

Aunque se necesitan más estudios con otros métodos y una metodología más sólida para perfeccionar el resultado del trabajo, teniendo en cuenta los resultados del proceso de selección de características y la clasificación, creemos que el resultado de este trabajo es un paso prometedor hacia una detección rápida y temprana de la EP.

En primer lugar, los resultados del proceso de selección de características sugieren que GDS, SCOPA-AUT, UPSIT y MoCA son las pruebas más informativas para discriminar los EPI de los CS, mientras que las pruebas ESS, SDMT, BJLOT y HVLTR son probablemente más informativas para otras cualidades de los EP.

Además, el haber obtenido los mejores resultados con los clasificadores SVM construidos sobre la base de datos BDI + SCC, sugieren, que si el objetivo es discriminar entre EPI y CS, ni siquiera sería necesario realizar el test completo o rellenar el cuestionario completo en el caso de GDS, SCOPA-AUT, UPSIT y MoCA. Bastaría con realizar la pregunta sobre energía en el cuestionario GDS; las preguntas sobre salivación, estreñimiento, pérdidas de orina, incluyendo la toma de medicación para ello, en el cuestionario SCAU; la prueba sobre la capacidad olfativa (UPSIT) completa y las pruebas correspondientes a la atención, fluidez verbal y recuerdo en MoCA.

VII. Conclusiones y trabajos futuros

Se estudia el problema de la detección precoz de la enfermedad de Parkinson, basándose en los síntomas no motores y buscando el subconjunto de elementos más adecuado. La selección de variables mediante el método SCC ha demostrado ser adecuada porque reduce considerablemente el número de elementos. Además, se propone una forma de agrupar estos elementos semánticamente. Para comprobar la eficacia de la propuesta, se evalúan varios algoritmos. Todos ellos alcanzan un valor de precisión superior al 81 % y un Valor-F superior al 86 %. El resultado no solo es satisfactorio por las métricas conseguidas, sino también porque el número de variables puede reducirse considerablemente, seleccionando los elementos más relevantes (13 variables) o agrupando algunas de ellas semánticamente (4 variables).

Como trabajos futuros se plantea, por un lado, crear una base de datos intermedias que agrupe algunas de las variables individuales, con el propósito de conseguir un mejor resultado utilizando pocos items. Por otro lado, hacer un estudio longitudinal utilizando las variables no motoras para evaluar si los síntomas motores precoces pueden servir como biomarcador para predecir el pronóstico de la enfermedad. Además, en ambos propósitos se sumarán modelos explicativos, como árboles de clasificación e inducción de reglas.

Acknowledgment

Este trabajo ha sido financiado por el Gobierno Vasco (ADIAN, IT-980-16); por el Ministerio de Economía y Competitividad del Gobierno español y el FEDER (PhysComp, TIN2017-85409-P); y por la Agencia Estatal de Investigación (AEI, España) en el marco del convenio de subvención No RED2018-102312-T (IA-Biomed).

Referencias

[1] A. Rahmim, P. Huang, N. Shenkov, S. Fotouhi, E. Davoodi-Bojd, L. Lu, Z. Mari, H. Soltanian-Zadeh, and V. Sossi, "Improved prediction of outcome in parkinson's disease using radiomics analysis of longitudinal dat spect images," *NeuroImage: Clinical*, vol. 16, pp. 539–544, 2017.

[2] F. P. Oliveira, D. B. Faria, D. C. Costa, M. Castelo-Branco, and J. M. R. Tavares, "Extraction, selection and comparison of features for an effective automated computer-aided diagnosis of parkinson's disease based on [123 i] fp-cit spect images," *European journal of nuclear medicine and molecular imaging*, vol. 45, no. 6, pp. 1052–1062, 2018.

[3] Z. K. Senturk, "Early diagnosis of parkinson's disease using machine learning algorithms," *Medical hypotheses*, vol. 138, p. 109603, 2020.

[4] C. R. Pereira, D. R. Pereira, S. A. Weber, C. Hook, V. H. C. de Albuquerque, and J. P. Papa, "A survey on computer-assisted parkinson's disease diagnosis," *Artificial intelligence in medicine*, vol. 95, pp. 48–63, 2019.

[5] G. Solana-Lavalle, J.-C. Galán-Hernández, and R. Rosas-Romero, "Automatic parkinson disease detection at early stages as a pre-diagnosis tool by using classifiers and a small set of vocal features," *Biocybernetics and Biomedical Engineering*, vol. 40, no. 1, pp. 505–516, 2020.

[6] K. Yasaka, K. Kamagata, T. Ogawa, T. Hatano, H. Takeshige-Amano, K. Ogaki, C. Andica, H. Akai, A. Kunimatsu, W. Uchida *et al.*, "Parkinson's disease: deep learning with a parameter-weighted structural connectome matrix for diagnosis and neural circuit disorder investigation," *Neuroradiology*, pp. 1–12, 2021.

[7] A. Murueta-Goyena, A. Andikoetxea, J. C. Gómez-Esteban, and I. Gabilondo, "Contribution of the gabaergic system to non-motor manifestations in premotor and early stages of parkinson's disease," *Frontiers in pharmacology*, vol. 10, p. 1294, 2019.

[8] N. Amoroso, M. La Rocca, A. Monaco, R. Bellotti, and S. Tangaro, "Complex networks reveal early mri markers of parkinson's disease," *Medical image analysis*, vol. 48, pp. 12–24, 2018.

[9] A. J. Hughes, S. E. Daniel, L. Kilford, and A. J. Lees, "Accuracy of clinical diagnosis of idiopathic parkinson's disease: a clinico-pathological study of 100 cases." *Journal of neurology, neurosurgery & psychiatry*, vol. 55, no. 3, pp. 181–184, 1992.

[10] S. G. Alonso, I. de la Torre-Díez, S. Hamrioui, M. López-Coronado, D. C. Barreno, L. M. Nozaleda, and M. Franco, "Data mining algorithms and techniques in mental health: A systematic review," *Journal of medical systems*, vol. 42, no. 9, pp. 1–15, 2018.

[11] W. Gulbinat, "What is the role of who as an intergovernmental organisation," *The coordination of telematics in healthcare. World Health Organisation Geneva, Switzerland*, 1997.

[12] E. Adeli, F. Shi, L. An, C.-Y. Wee, G. Wu, T. Wang, and D. Shen, "Joint feature-sample selection and robust diagnosis of parkinson's disease from mri data," *NeuroImage*, vol. 141, pp. 206–219, 2016.

[13] M. Wenzel, F. Milletari, J. Krüger, C. Lange, M. Schenk, I. Apostolova, S. Klutmann, M. Ehrenburg, and R. Buchert, "Automatic classification of dopamine transporter spect: deep convolutional neural networks can be trained to be robust with respect to variable image characteristics," *European journal of nuclear medicine and molecular imaging*, vol. 46, no. 13, pp. 2800–2811, 2019.

[14] A. Llera, I. Huertas, P. Mir, and C. F. Beckmann, "Quantitative intensity harmonization of dopamine transporter spect images using gamma mixture models," *Molecular imaging and biology*, vol. 21, no. 2, pp. 339–347, 2019.

[15] G. Singh, M. Vadera, L. Samavedham, and E. C.-H. Lim, "Machine learning-based framework for multi-class diagnosis of neurodegenerative diseases: A study on parkinson's disease," *IFAC-PapersOnLine*, vol. 49, no. 7, pp. 990–995, 2016.

[16] F. Cavallo, A. Moschetti, D. Esposito, C. Maremmani, and E. Rovini, "Upper limb motor pre-clinical assessment in parkinson's disease using machine learning," *Parkinsonism & related disorders*, vol. 63, pp. 111–116, 2019.

[17] J. S. Almeida, P. P. Rebouças Filho, T. Carneiro, W. Wei, R. Damaševičius, R. Maskeliūnas, and V. H. C. de Albuquerque, "Detecting parkinson's disease with sustained phonation and speech signals using machine learning techniques," *Pattern Recognition Letters*, vol. 125, pp. 55–62, 2019.

[18] O. Yaman, F. Ertam, and T. Tuncer, "Automated parkinson's disease recognition based on statistical pooling method using acoustic features," *Medical hypotheses*, vol. 135, p. 109483, 2020.

[19] M. Wodzinski, A. Skalski, D. Hemmerling, J. R. Orozco-Arroyave, and E. Nöth, "Deep learning approach to parkinson's disease detection using voice recordings and convolutional neural network dedicated to image classification," in *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2019, pp. 717–720.

[20] K. Marek, D. Jennings, S. Lasch, A. Siderowf, C. Tanner, T. Simuni, C. Coffey, K. Kieburtz, E. Flagg, S. Chowdhury *et al.*, "The parkinson progression marker initiative (ppmi)," *Progress in neurobiology*, vol. 95, no. 4, pp. 629–635, 2011.

[21] K. Marek, S. Chowdhury, A. Siderowf, S. Lasch, C. S. Coffey, C. Caspell-Garcia, T. Simuni, D. Jennings, C. M. Tanner, J. Q. Trojanowski *et al.*, "The parkinson's progression markers initiative (ppmi)–establishing a pd biomarker cohort," *Annals of clinical and translational neurology*, vol. 5, no. 12, pp. 1460–1477, 2018.

[22] M. W. Johns, "A new method for measuring daytime sleepiness: the epworth sleepiness scale," *sleep*, vol. 14, no. 6, pp. 540–545, 1991.

[23] J. A. Yesavage, T. L. Brink, T. L. Rose, O. Lum, V. Huang, M. Adey, and V. O. Leirer, "Development and validation of a geriatric depression screening scale: a preliminary report," *Journal of psychiatric research*, vol. 17, no. 1, pp. 37–49, 1982.

[24] M. Visser, J. Marinus, A. M. Stiggelbout, and J. J. Van Hilten, "Assessment of autonomic dysfunction in parkinson's disease: the scopa-aut," *Movement disorders: official journal of the Movement Disorder Society*, vol. 19, no. 11, pp. 1306–1312, 2004.

[25] A. Smith, *Symbol digit modalities test*. Western Psychological Services Los Angeles, 1973.

[26] A. L. Benton, N. R. Varney, and K. d. Hamsher, "Visuospatial judgment: A clinical test," *Archives of neurology*, vol. 35, no. 6, pp. 364–367, 1978.

[27] R. L. Doty, P. Shaman, C. P. Kimmelman, and M. S. Dann, "University of pennsylvania smell identification test: a rapid quantitative olfactory function test for the clinic," *The Laryngoscope*, vol. 94, no. 2, pp. 176–178, 1984.

[28] Z. S. Nasreddine, N. A. Phillips, V. Bédirian, S. Charbonneau, V. Whitehead, I. Collin, J. L. Cummings, and H. Chertkow, "The montreal cognitive assessment, moca: a brief screening tool for mild cognitive impairment," *Journal of the American Geriatrics Society*, vol. 53, no. 4, pp. 695–699, 2005.

[29] R. H. Benedict, D. Schretlen, L. Groninger, and J. Brandt, "Hopkins verbal learning test–revised: Normative data and analysis of inter-form and test-retest reliability," *The Clinical Neuropsychologist*, vol. 12, no. 1, pp. 43–55, 1998.

[30] Y. Zhang, S. Li, T. Wang, and Z. Zhang, "Divergence-based feature selection for separate classes," *Neurocomputing*, vol. 101, pp. 32–42, 2013.

[31] M. A. Hall, "Correlation-based feature selection for machine learning," 1999.

[32] P. Langley, W. Iba, K. Thompson *et al.*, "An analysis of bayesian classifiers," in *Aaai*, vol. 90. Citeseer, 1992, pp. 223–228.

[33] J. Joyce, "Bayes' theorem," 2003.

[34] A. H. Jahromi and M. Taheri, "A non-parametric mixture of gaussian naive bayes classifiers based on local independent features," in *2017 Artificial Intelligence and Signal Processing Conference (AISP)*. IEEE, 2017, pp. 209–212.

[35] S. Suthaharan, "Support vector machine," in *Machine learning models and algorithms for big data classification*. Springer, 2016, pp. 207–235.

[36] T. Hastie, R. Tibshirani, and J. Friedman, *The elements of statistical learning: data mining, inference, and prediction*. Springer Science & Business Media, 2009.

[37] S. Sharma, "Activation functions in neural networks," *towards data science*, vol. 6, 2017.

[38] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747*, 2016.

[39] G. Cybenko, "Approximation by superpositions of a sigmoidal function," *Mathematics of control, signals and systems*, vol. 2, no. 4, pp. 303–314, 1989.

[40] N. S. Altman, "An introduction to kernel and nearest-neighbor nonparametric regression," *The American Statistician*, vol. 46, no. 3, pp. 175–185, 1992.

[41] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[42] M. Xu, P. Watanachaturaporn, P. K. Varshney, and M. K. Arora, "Decision tree regression for soft classification of remote sensing data," *Remote Sensing of Environment*, vol. 97, no. 3, pp. 322–336, 2005.

[43] S. Abney, "Bootstrapping," in *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, 2002, pp. 360–367.

[44] S. Purushotham and B. Tripathy, "Evaluation of classifier models using stratified tenfold cross validation techniques," in *International Conference on Computing and Communication Systems*. Springer, 2011, pp. 680–690.

[45] C. Sammut and G. I. Webb, *Encyclopedia of machine learning*. Springer Science & Business Media, 2011.

Detecting risk of self-harm on Reddit using ensemble traditional and deep learning models

1st Elena Campillo-Ageitos
NLP & IR Group,
Lenguajes y Sistemas Informáticos (UNED)
Madrid, Spain
ecampillo@lsi.uned.es

2nd Juan Martinez-Romo
NLP & IR Group,
Lenguajes y Sistemas Informáticos (UNED)
Instituto Mixto de Investigación - Escuela Nacional de Sanidad (IMIENS)
Madrid, Spain
juaner@lsi.uned.es

3rd Lourdes Araujo
NLP & IR Group,
Lenguajes y Sistemas Informáticos (UNED)
Instituto Mixto de Investigación - Escuela Nacional de Sanidad (IMIENS)
Madrid, Spain
lurdes@lsi.uned.es

***Abstract*—Self-harm is the act of causing bodily harm to oneself without suicidal intent. It is a maladaptive coping mechanism, observed mostly in young people, that has been linked to serious mental health problems such as depression and anxiety. It can lead to serious risks such as suicide, and needs to be monitored to improve the help we give these people. Research suggests that the emotional well-being of a person can be detected by the way they talk or communicate through writing. Social media provides a massive source of user-generated natural text to train detection systems. Early detection is crucial for mental health problems, since early intervention is critical. Annotated datasets are needed to train supervised systems. The shared task eRisk proposed a self-harm early risk detection dataset extracted from Reddit. Using this, we propose a classification system based on ensemble models that combine traditional machine learning systems and deep learning systems such as CNNs and LSTMs. Features are obtained using TF-IDF and embeddings. Our results show that ensemble models show promise in the field of mental health detection, but further work needs to be done to develop competent health recommendation systems.**

***Index Terms*—self-harm, early risk detection, social media, ensemble models, deep learning, CNN, LSTM, bidirectional LSTM**

I. Introduction

A correct diagnosis in mental health is very important, but early diagnosis is critical for the well-being and prognosis of the patient. An early intervention can prevent the illness from developing further, and can prevent some of the associated risk behaviours: drug abuse, social exclusion, suicide, and even death. Given the severity of these risks, early detection and intervention are critical.

One of the lesser-known risk behaviours is self-harm, which affects mostly young people [1]. According to the Oxford dictionary, self-harm is "the practice of deliberately injuring yourself, for example by cutting yourself, as a result of having serious emotional or mental problems" [2]. It is the act of causing bodily harm to oneself with no suicidal intent. Self-harmers engage in cutting their own skin, burning or hair pulling. It is a maladaptive form of coping [3] with underlying mental illnesses, such as depression and anxiety [4], and can lead to severe consequences, such as accidental suicide. Detecting people who engage in self-harm or who are at risk of turning to self-harm can improve the medical attention they receive.

The way people talk and write conveys not only the meaning of the message, but the emotional state of the speaker. Natural Language Processing (NLP) can be used to extract information about a person's emotional health from written texts. With this information, recommendation and alert systems can be developed to assist medical practitioners to better detect at-risk patients.

The past ten years have seen the rise of social media. Sites such as Facebook, Twitter, Instagram and Reddit have amassed users in the range of thousands of millions from all around the world. These users create and post content: pictures and text. This text can be used to monitor their mental health and their risk of falling into risk activities by using NLP techniques.

This paper presents a preliminary study on the application of ensemble deep learning models for the detection and classification of at-risk users on an existing social media dataset.

The following sections are ordered as follows: 1) A review of previous work done on the field, 2) A description of the dataset, 3) A description of the models, 4) experiment results and discussion, and 5) conclusions and future work.

II. Previous Work

It is said that what people say, and how they say it, conveys the emotional state of the speaker. Linguistic patterns can be used to determine whether a person's mental wellbeing is being compromised [5]. Not only the meaning of the words used is important, but even particles can convey emotional state and cognitive styles [6].

Since the rise of social media, a wide amount of self-published content was made available for researchers to study. Data from Twitter, Facebook, Instagram, Reddit and Weibo are often used. Many studies have been done on detecting mental illness on social media data. Most recently, Chiu et al. [7] measured the aggregated depression score of posts from Instagram accounts to detect depressive users, analysing the pictures and the time intervals between each post. Birnbaum et al. [8] chose

Facebook data for their research: they detected Schizophrenia and mood disorders using posts and images, and found that ill users used more perception words, more swear words, and were more likely to express negative emotions. Fatima et al. [9] predicted postpartum depression using absolutist words.

Studies performed specifically on self-harm are scarce. Some studies show that self-harmers have common patterns on high negative affectivity, and prove that it is a maladaptive coping strategy [10], [11]. Some work can be found on studying self-harm behaviour in social media, but it mostly focuses on posting patterns and behaviours [12], [13]. These researchers found that self-harmers post on social media in different patterns than control users.

There are several examples in the literature of NLP techniques used to detect mental health risks. Deep learning shows promise, but more research needs to be done [14]–[16].

Most datasets available for mental health detection rely on diagnosis to classify healthy people from people at risk. However, the goal is to detect markers, risks and hints of mental problems before they develop, so medical practitioners can intervene as soon as possible, and improve prognosis or even prevent it from happening. In this context, shared tasks such as eRisk [17] have been proposed. This task focuses on early detection of several mental health problems, such as depression, anorexia, and self-harm on temporal data extracted from Reddit.

The 2020 eRisk task [18] proposed two different subtasks: Task 1 focused on early detection of signs of self-harm. The dataset for this task is a collection of chronological posts written by different users on Reddit. The goal of this task was to evaluate writings one at a time and give a prediction on whether a subject showed signs of self-harm as early as possible. Some successful systems which showed promising results were team iLab in 2020 [19]. iLab proposed classifiers based on BERT (Bidirectional Encoder Representations from Transformers), obtaining best results for that years' task. Deep learning models show the best results, but they lack explainability. They are also much slower than traditional machine learning models.

Proposed solutions for the eRisk task have been very varied. Most participants apply at least some form of deep learning, but we can also see the use of ensemble models: the researchers from team HILDESHEIM [20] applied an ensemble system where they combined a temporal difference detection system, a sentiment analysis system, a distributional semantics system, and a neuronal network system; some teams like PRHLT-UPV [21] applied linguistic, emotion and sentiment features to several neural network classifiers. But some participants also applied traditional machine learning systems, like BIOINFO@UAVR [22]. The researchers proposed an SVM classifier based on Bag of Words and TF-IDF based features, as well as paragraph vectors based on Doc2vec, applied to *Adaboost* classifier.

III. Dataset Description

In this section, we present the dataset we used for this preliminary study, and explore some of its characteristics.

The dataset is an early risk detection dataset presented by Losada et al. at [23]. It was developed for a CLEF shared task for early risk detection of mental health concerns, such as anorexia, gambling, and self-harm. The data was extracted from Reddit, and it presents a collection of users and messages. Each document corresponds to a different user, and in it, there is an arbitrary amount of messages or posts (submissions done to Reddit). They are ordered from oldest to earliest. For more information on the dataset, check [23].

The format of a subjects' document is an XML as follows:

```
[
    {
        "id": 18752,
        "number": 0,
        "nick": "subject3798",
        "redditor": 18702,
        "title": "...",
        "content": "...",
        "date": "..."
    },
    {
        "id": 18772,
        "number": 1,
        "nick": "subject3798",
        "redditor": 18702,
        "title": "...",
        "content": "...",
        "date": "..."
    },
    ...
]
```

Where nick, title and content are relevant to this study: 1) Nick: an alias that identifies the user, 1) title: the title of the post the user made, which can exist or not, 2) content: the content of the post. A different document contains a relation of user nicks and golden truths.

As we can see, there is a temporal and sequential component to the data in this dataset. Our goal is to classify the user, but each user has a different amount of individual datum, sequentially ordered. The challenge in the eRisk shared task is to observe the messages sequentially, one at a time, from oldest to newest, and classify the user as positive or negative as early as possible. For this preliminary study, we decided to forego temporality, and joined each users' posts into a single document. We manipulated the dataset, so the format became as follows:

```
[
    {
        \nick": \subject3798",
        \text": title1 + content1
        + title2 + content2 + ...
```

```
    },
    {
        "nick": "subject7495",
        "text": title1 + content1
        + title2 + content2 + ...
    },
    ...
]
```

This way, the problem turns into a simpler document classification problem, where each document corresponds to an users' writings.

Each year the eRisk organizers release an updated dataset. This particular dataset belongs to the 2020 eRisk shared task 2, early detection of signs of self-harm. The dataset is a collection of two groups of Reddit users: those who have explicitly said in the platform that they engage in self-harm, and a control group. Those messages that were used to identify self-harmers were removed from the collection by the organizers. The data came in two groups: train set and test set. We decided to use this division for our train and test division.

The texts written in these documents are not formal texts; they are written by people of all ages and areas on the social media site known as Reddit. Grammar and spelling are not always going to be correct. Emojis are going to be used; emoticons are going to be used; links, weird expressions, capitals, many vowels, too few vowels, etc. These carry meaning on themselves (for example, writing a text in all caps usually conveys that the person is shouting, excited, etc.). This can be a challenge for deep learning systems, because we need to convert text to numbers, and for that we use embeddings.

Embeddings are a vector representation of words that attempt to represent proximity between words. Most pre-trained embeddings are not prepared to deal with Internet language, only with normal text and normal words.

Furthermore, the proof we have that any user engages in self-harm is self-report. They wrote somewhere on Reddit at some point that they harm themselves. We find two problems with this: first, while it is unlikely somebody would lie about engaging in self-harm, the possibility is there. The second problem is related to the temporal nature of the dataset. A person is not a monolith in time, and just because they do self-harm now, does not mean that they have always done it, and always will. This dataset assumes that every post a self-harmer has written was done while doing self-harm or while showing early signs of future self-harm, and this will not always be true. Therefore, we believe there is a certain noise characteristic to this dataset.

TABLE I: eRisk 2020 dataset users breakdown on positive and negative groups, and train and test groups.

eRisk 2020 data	Train	Test	Total
Positive users	41	104	145
Negative users	299	319	618
All users	340	423	763

Table I shows a summary of the eRisk dataset. There are 763 users in total, divided into 145 positive users and 618 negative users. This is a percentage of 19% positive users, which makes the data highly unbalanced. The data was divided into a training and testing group, with sizes 340 and 423 respectively. The following statistics are performed on all the data.

TABLE II: eRisk 2020 dataset average and median text length for all documents of each user.

eRisk text length	*Text length mean*	*Text length median*
Positive users	36366.18	17547
Negative users	67699.43	28245.5
All users	61744.88	24998

Table II shows information about the length of users' texts, and the number of words. As we can see, negative users show the same distribution as all users, but positive users follow a particular pattern. Texts are, in general, shorter for positive users.

In general, either positive users write shorter posts than negative users, or they write less often. But we can see there are significant differences between the behaviour of positive and negative users, and we hope those can be exploited with our system.

IV. System Description

In this section, we present how we prepared the data for classification, the machine and deep learning models we studied, and the ensemble models.

A. Data Preprocessing

We preprocessed the data in the following steps: first, we joined every text post for each user into a single document. Next, we cleaned the text using the Python library *RedditTokenizer* [24] to remove links, hyperparameters, user mentions, emojis, etc. We also performed some typical preprocessing, such as lowercasing, removing numbers, etc.

One of the challenges of this dataset is, as mentioned earlier, the difference between the number of self-harm (positive) users and control (negative) users. We applied random oversampling with the Python library *imblearn.over_sampling* to the positive users of the training group. This artificially increased the amount of positive users by duplicating a random selection of them. Afterwards, the size of the positive and negative classes were each 299 users.

Finally, the texts were transformed into features. This step was different depending on the type of model. For deep learning models, we applied embeddings; for machine learning models, we applied TF-IDF (term frequency–inverse document frequency).

a) Embeddings: We used Stanford's pre-trained GloVe [25] Wikipedia 2014 and Gigaword 5 100d word embeddings. We converted every document (users' collection of comments) into an embedding vector. Each vector was padded to a size of 50,000. We chose this size because the documents for each user can be pretty large, and we wanted to ensure that the length of the overall document was preserved while still

applying *padding*, since we observed a difference in text length between positive and negative users.

b) TF-IDF: We applied Scikit-Learn's *TfidfVectorizer*. The vectorizer was trained on the positive users of the training group with 50,000 max features. Then, the inverse term frequencies were calculated for each word in each document of the train and test groups using this vectorizer.

B. Traditional Machine Learning Systems

We trained three different traditional machine learning systems: 1) Multinomial naive bayes (NB), 2) non-linear support vector machine (SVM) and 3) linear support vector machine (LiSVM). These were implemented using the Python library *Scikit-Learn*[1].

The *Scikit-Learn* classes used were *MultinomialNB* for NB, *SVC* for SVM, and *LinearSVC* for LiSVM. All were applied using default parameters.

C. Deep Learning Systems

For this preliminary study, we chose two different deep learning algorithms: 1) A convolutional neural network (CNN) and 2) a long short-term memory with bidirectionality (BiLSTM). Two BiLSTM models were prepared with different layer configurations, so we had three different deep Learning models: 1) CNN, 2) BiLSTM1 and 3) BiLSTM2. Figure 1 shows the layer configuration of the models.

D. Ensemble Systems

According to the preliminary evaluations we performed on the machine learning and deep learning models, we assembled four different Ensemble models with combinations of the previously mentioned models. These are:

- Ensemble 1 (E1): LiSVM+NB+CNN
- Ensemble 2 (E2): SVM+NB+BiLSTM1
- Ensemble 3 (E3): LiSVM+NB+BiLSTM1
- Ensemble 4 (E4): LiSVM+NB+BiLSTM2

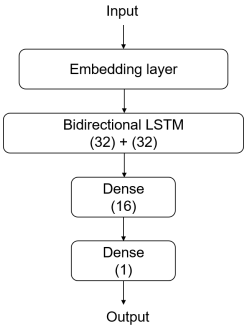
These four Ensemble models were designed as hard voting ensembles. That is, each simple model classified the data separately and predicted (voted) one class, and the Ensemble model predicted the class with the most votes.

E. Training

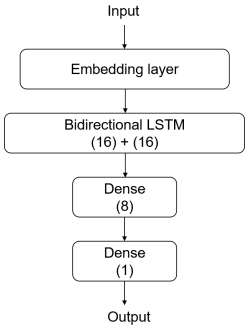
Training was performed on each model before assembling the Ensemble models. Deep learning models (CNN, BiLSTM1, BiLSTM2) were trained on embeddings, and machine learning models (BN, SVM, LiSVM) were trained on TF-IDF features.

For the deep learning systems, we trained with batch sizes of 16 and 10 epochs. The data was shuffled before being sent to training. The random seed for the *Numpy* and *Tensorflow* backend was set as 42 in order to achieve reproducible results. During training, 20% of the data was used for model validation.

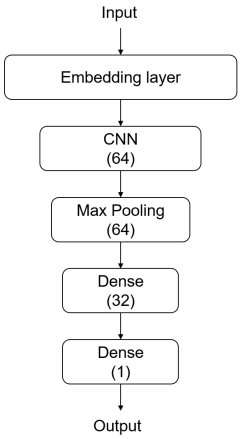
[1]https://scikit-learn.org/



(a) BiLSTM model 1.



(b) BiLSTM model 2.



(c) CNN model.

Fig. 1: Deep learning models.

V. Results and Discussion

The evaluation measures we use to compare the models are the F-measure of the positive class, recall of the positive class, and the *macro-average F-measure* of both classes. We chose F-measure because the classes are unbalanced. Recall is important as well because we are working in the medical field. In this case, recall is more important than precision. Classifying an at-risk patient as negative has very serious consequences, so we want to make sure that we miss as few as possible. On the other hand, accidentally classifying negative users as positive is a waste of resources because they have to be monitored, but the risk is less serious.

Table III shows the results of the machine learning, deep learning, and ensemble models. The ensemble model with the best results is bolded.

a) Traditional machine learning systems: Surprisingly, the Multinomial Naive Bayes (NB) performed best out of the three systems with a macro f score of 0.83. All three achieved good results. Recall was moderate, only obtaining a maximum of 0.64.

b) Deep learning systems: These systems performed comparatively worse to the traditional machine learning systems. The best model, CNN, obtained a *macro-average F-measure* of 0.72 and a recall measure of 0.39.

TABLE III: Results of the evaluation using the eRisk 2020 dataset of the proposed traditional, Deep and Ensemble learning models. Measures shown are F-measure for the positive class, *macro-average F-measure*, and recall for the positive class.

Evaluation results	***F1 (class 1)***	***macro F1***	***R (class 1)***
Machine learning models			
NB	0.74	0.83	0.64
SVM	0.68	0.80	0.57
LiSVM	0.70	0.81	0.62
Deep learning models			
CNN	0.54	0.72	0.39
BiLSTM1	0.51	0.68	0.47
BiLSTM2	0.52	0.68	0.54
Ensemble models			
E1 (LiSVM+NB+CNN)	0.72	0.82	0.60
E2 (SVM+NB+BiLSTM1)	0.74	0.84	0.62
E3 (LiSVM+NB+BiLSTM1)	**0.75**	**0.84**	**0.64**
E4 (LiSVM+NB+BiLSTM3)	0.72	0.82	0.62

c) Ensemble systems: Overall, the E3 system performed the best out of every system. It obtained a macro f score of 0.84 and recall of 0.64. It is a small improvement over the NB system, which is used as part of this ensemble model. It is interesting to note that the ensemble models that used the best deep learning system (CNN) obtained worse results than E3, which used BiLSTM1.

In general, we can see in these results that the unbalanced nature of the data supposes a big challenge. The deep learning models (CNN, BiLSTM1, BiLSTM2) obtained poor results in F-measure and recall. The traditional systems and the ensemble systems obtained better results, but they still can be improved. Even after oversampling, the measures for the positive class are more difficult to optimize, perhaps because the oversampling was only performed on the training group.

It is important to note that the oversampling done on the training data was big, and thus the results should be taken with that in consideration. More reliable results would need a balanced dataset, or a more advanced technique for balancing the training data.

VI. Conclusions and Future Work

In this preliminary study we evaluated an early risk detection dataset of self-harm on Reddit and classified it with ensemble models made with a combination of simple machine learning models, convolutional neural networks and bidirectional LSTMs. Our results showed that the ensemble models improve results over the individual models. However, more work needs to be done to perfect them and find the best configuration to improve precision and recall. Furthermore, the dataset is challenging, since the negative class is overrepresented over the positive class.

The next step we propose is to implement more complex deep learning models. Different embeddings such as *FastVect* can also be applied, and even trained on the eRisk dataset. Another line of work to continue working on is to explore solutions to the unbalanced dataset. We applied simple random oversampling, but more advanced techniques could be applied.

Acknowledgment

This work has been partially supported by the Spanish Ministry of Science and Innovation within the DOTT-HEALTH Project (MCI/AEI/FEDER, UE) under Grant PID2019-106942RB-C32, as well as project EXTRAE II (IMIENS 2019) and the research network AEI RED2018-102312-T (IA-Biomed).

References

[1] S. Gluck, "Self-Injurers and Their Common Personality Traits — HealthyPlace," 2020-02-12. [Online]. Available: https://www.healthyplace.com/abuse/self-injury/self-injurers-and-their-common-personality-traits

[2] "self-harm noun - Definition, pictures, pronunciation and usage notes — Oxford Advanced American Dictionary at OxfordLearnersDictionaries.com," 2021-05-25. [Online]. Available: https://www.oxfordlearnersdictionaries.com/definition/american_english/self-harm

[3] S. P. Lewis and D. A. Santor, "Self-Harm Reasons, Goal Achievement, and Prediction of Future Self-Harm Intent," *The Journal of Nervous and Mental Disease*, vol. 198, no. 5, pp. 362–369, may 2010. [Online]. Available: http://journals.lww.com/00005053-201005000-00008

[4] M. M. Greaves, "A Corpus Linguistic Analysis of Public Reddit and Tumblr Blog Posts on Non-Suicidal Self-Injury, An abstract," 2018.

[5] J. W. Pennebaker and A. Graybeal, "Patterns of natural language use: Disclosure, personality, and social integration," *Current Directions in Psychological Science*, vol. 10, no. 3, pp. 90–93, 2001.

[6] J. W. Pennebaker, M. R. Mehl, and K. G. Niederhoffer, "Psychological aspects of natural language use: Our words, our selves," *Annual Review of Psychology*, vol. 54, no. 1, pp. 547–577, 2003.

[7] C. Y. Chiu, H. Y. Lane, J. L. Koh, and A. L. Chen, "Multimodal depression detection on instagram considering time interval of posts," *Journal of Intelligent Information Systems*, vol. 56, no. 1, pp. 25–47, 2021.

[8] M. L. Birnbaum, R. Norel, A. Van Meter, A. F. Ali, E. Arenare, E. Eyigoz, C. Agurto, N. Germano, J. M. Kane, and G. A. Cecchi, "Identifying signals associated with psychiatric illness utilizing language and images posted to Facebook," *NPJ schizophrenia*, vol. 6, no. 1, p. 38, 2020.

[9] I. Fatima, B. U. D. Abbasi, S. Khan, M. Al-Saeed, H. F. Ahmad, and R. Mumtaz, "Prediction of postpartum depression using machine learning techniques from social media text," *Expert systems*, vol. 36, no. 4, p. n/a, 2019.

[10] I. Baetens, L. Claes, J. Muehlenkamp, H. Grietens, and P. Onghena, "Non-Suicidal and Suicidal Self-Injurious Behavior among Flemish Adolescents: A Web-Survey," *Archives of Suicide Research*, vol. 15, no. 1, pp. 56–67, 2011. [Online]. Available: https://doi.org/10.1080/13811118.2011.540467

[11] K. A. Nicolai, M. D. Wielgus, and A. Mezulis, "Identifying Risk for Self-Harm: Rumination and Negative Affectivity in the Prospective Prediction of Nonsuicidal Self-Injury," *Suicide and Life-Threatening Behavior*, vol. 46, no. 2, pp. 223–233, apr 2016. [Online]. Available: http://doi.wiley.com/10.1111/sltb.12186

[12] P. A. Cavazos-Rehg, M. J. Krauss, S. J. Sowles, S. Connolly, C. Rosas, M. Bharadwaj, R. Grucza, and L. J. Bierut, "An analysis of depression, self-harm, and suicidal ideation content on Tumblr," *Crisis*, vol. 38, no. 1, pp. 44–52, 2017. [Online]. Available: https://psycnet.apa.org/record/2016-36501-001

[13] C. Emma Hilton, "Unveiling self-harm behaviour: what can social media site Twitter tell us about self-harm? A qualitative exploration," *Journal of Clinical Nursing*, vol. 26, no. 11-12, pp. 1690–1704, jun 2017. [Online]. Available: http://doi.wiley.com/10.1111/jocn.13575

[14] M. M. Agüero-Torales, J. I. Abreu Salas, and A. G. López-Herrera, "Deep learning and multilingual sentiment analysis on social media data: An overview," *Applied soft computing*, vol. 107, 2021.

[15] J. Kim, J. Lee, E. Park, and J. Han, "A deep learning model for detecting mental illness from user content on social media," *Scientific reports*, vol. 10, no. 1, p. 11846, 2020.

[16] Y. T. Wang, H. H. Huang, and H. H. Chen, "A neural network approach to early risk detection of depression and anorexia on social media text," in *CEUR Workshop Proceedings*, vol. 2125, 2018.

[17] "CLEF eRisk: Early risk prediction on the internet," 2020-01-28. [Online]. Available: https://early.irlab.org/server.html https://early.irlab.org/

[18] D. E. Losada, F. Crestani, and J. Parapar, "Overview of eRisk 2020: Early Risk Prediction on the Internet," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 12260 LNCS, pp. 272–287, 2020.

[19] R. Martínez-Castaño, A. Htait, L. Azzopardi, and Y. Moshfeghi, "Early risk detection of self-harm and depression severity using BERT-based transformers : iLab at CLEF eRisk 2020," pp. 22–25, 2020.

[20] L. Achilles, M. Kisselew, J. Sch, and R. Koelle, "Using Surface and Semantic Features for Detecting Early Signs of Self-Harm in Social Media Postings," pp. 22–25.

[21] A.-s. Uban and P. Rosso, "Deep learning architectures and strategies for early detection of self-harm and depression level prediction," pp. 22–25.

[22] A. Trifan and P. Salgado, "BioInfo @ UAVR at eRisk 2020 : on the use of psycholinguistics features and machine learning for the classification and quantification of mental diseases," pp. 22–25, 2020.

[23] D. E. Losada and F. Crestani, "A Test Collection for Research on Depression and Language Use CLEF 2016, Évora (Portugal)," *Experimental IR Meets Multilinguality, Multimodality, and Interaction*, pp. 28–29, 2016. [Online]. Available: https://tec.citius.usc.es/ir/pdf/evora.pdf

[24] Erikavaris, "erikavaris/tokenizer: Tokenizer for Twitter and Reddit data," 2021-05-25. [Online]. Available: https://github.com/erikavaris/tokenizer

[25] J. Pennington, R. Socher, and C. D. Manning, "GloVe: Global Vectors for Word Representation," in *Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1532–1543. [Online]. Available: http://www.aclweb.org/anthology/D14-1162

The importance of interpretability and visualization in ML for medical applications

Alfredo Vellido
IDEAI-UPC Research Center, UPC BarcelonaTech, 08034 Barcelona, Spain
CIBER-BBN, Cerdanyola del Vallès, Barcelona, Spain
avellido@cs.upc.edu

***Abstract*—Many areas of science have made a sharp transition towards data-dependent methods, enabled by simultaneous advances in data acquisition and the development of networked system technologies. This is particularly clear in the life sciences, which can be seen as a perfect scenario for the use of machine learning to address problems in which more traditional data analysis approaches might struggle. But this scenario also poses some serious challenges. One of them is the lack interpretability and explainability for complex nonlinear models. In medicine and health care, not addressing such challenge might seriously limit the chances of adoption of these methods. In this summary paper, we pay specific attention to one of the ways in which interpretability and explainability can be addressed in this context: data and model visualization.**

***Index Terms*—Interpretability and explainability, Data Visualization, Machine Learning, medicine, health care, medical decision support systems.**

I. Introduction

The overabundance of data in the modern life sciences could be seen as a perfect scenario for the use of machine learning (ML), but comes accompanied by some far from trivial challenges [1]. One of them is model interpretability and explainability. In medicine and health care, where explainability is paramount and the societal impact is potentially high [2], such challenge might seriously limit the chances of adoption, in real practice, of computer-based systems that rely on opaque ML methods for data analysis. In this summary paper, we pay specific attention to one of the ways in which the challenge can be addressed: through techniques for data visualization. By doing so, we aim to stress the importance of considering the human factor when attempting to enhance model interpretability in general and the importance of integrating the medical expert in the process of developing strategies to guarantee the interpretability and explainability of medical data models.

II. Interpretable ML in medicine: a key to adoption

Data-dependence is bound to increase in medical practice, given the prominent place occupied by evidence-based medicine in the current agenda. The simultaneous creation of an information-rich medical environment and the development of techniques for knowledge extraction tailored to this domain, would seem to be a win-win situation for ML. But lack of model interpretability is a problem with obvious implications: if an ML-based Medical Decision Support System (MDSS) churns out decisions that cannot be described in comprehensible terms, an insurmountable barrier is raised between the MDSS and the human subjects. The medical expert could not trust to implement a decision that she or he cannot explain, whereas the patient might not trust experts that base their judgement on unexplainable computer outcomes. This means that formal frameworks for machine-human interaction pursuing interpretability and explainability are even more important in medicine than in other ambits of science, specially because there is a constellation of stakeholders in the health domain with possibly quite different explanation needs [3]. These frameworks should almost be considered as a pre-requisite in the development of ML-based MDSS.

III. Visualization as a problem in medicine

Visualization has been mentioned to play a central role as an interpretability tool for medicine and it is important to provide a formal framework for its use in this area. The human analyst has an active role in the interactive visualization framework proposed by Sacha and co-workers in [4], acting as a bridge between visual pattern discovery (mostly using ML tools) and knowledge validation by external experts. The importance of appraising the possible benefits of putting the "human-in-the-loop" is persuasively argued precisely as a validating actor in practical applications.

In real-world medicine, visual discovery is not always purely exploratory and, therefore, potentially interesting patterns obtained through visualization must be validated against expert knowledge from the domain. Often, this external assessment requires a committee of experts who, in turn, will provide feedback to the analyst that can help to redesign visualization experiments. This adds an extra layer of human subjectivity to the interpretation task through visualization. As a result, the framework must care not only about a cycle involving computer-based visual techniques and a human analyst, but also about a coupled cycle involving two human parts: the data analyst and the experts from the medical domain who provide the ultimate expert verification.

A detailed representation of this interpretability-through-visualization cycle can be found in Fig. 1. It involves requests from the medical experts to the data analyst, including: a) guarantees of interpretability and explainability that are adapted to the specific requirements of the medical problem; b) model compliance with clinical protocols and guidelines for

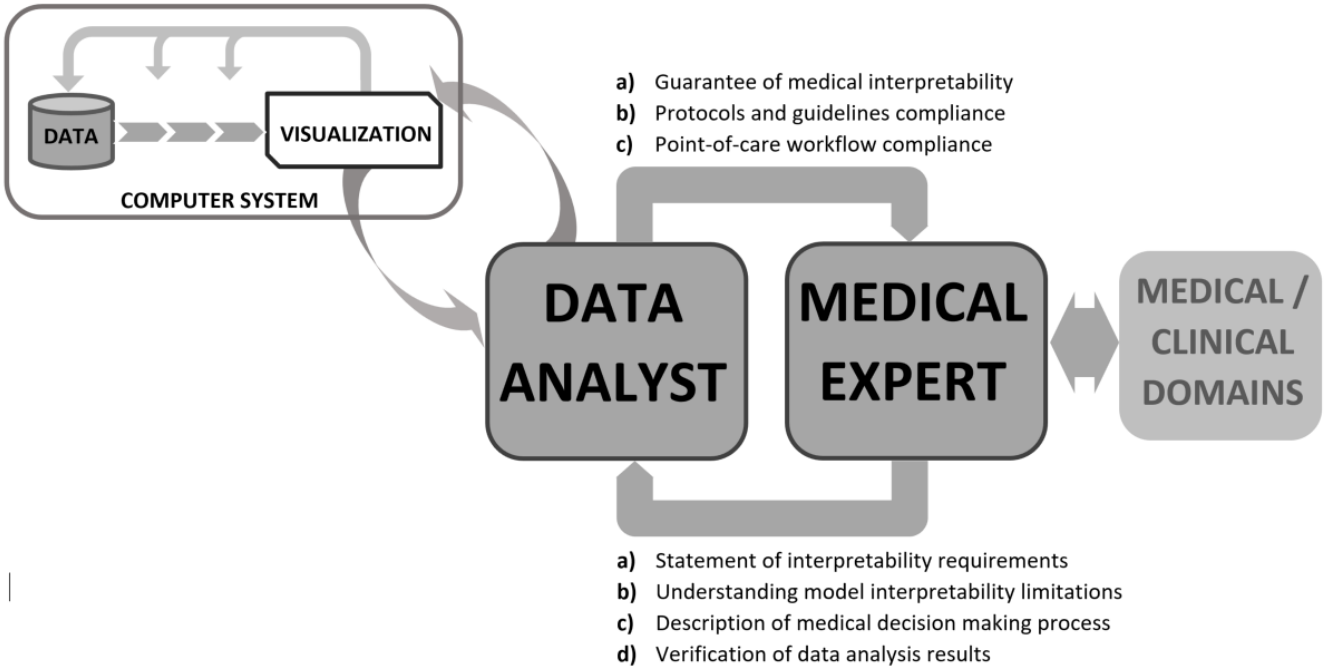


Fig. 1. Extension of the human analyst-computer ML interpretability cycle through interactive visualization proposed in [4] to account for a new sub-cycle of importance to the medical and health care domains. This new sub-cycle covers the necessary interaction between the human analyst, who must deliver data models that are interpretable and/or explainable from a medical viewpoint, and the medical expert, who must ensure that the data analyst is informed of the requirements that make interpretability valid from a medical standpoint. Arrows in the graphical depiction of the interaction between these two agents point from the agent that can deliver the interpretability item to the agent that requires it.

a given problem; c) model compliance with system-human interaction workflows at the point of care. It also involves requests from the data analyst to the medical experts, such as: a) a clear statement of the medical requirements concerning interpretability and explainability; b) a realistic understanding of the interpretability limitations and possibilities of the analytical models; c) a clear description of the real medical decision making process in place at the point of care; and d) a guarantee of verification of the data analysis results.

IV. CONCLUSIONS

The life sciences are at the avant-garde of an irreversible trend that is placing data at the heart of scientific discovery. Medicine and health care, at their own pace, are following suit. This is an unprecedented opportunity for ML, CI and related techniques for knowledge extraction from data. In this paper summary, we have argued that there are still many barriers to overcome before these techniques become mainstream. One of them is model interpretability and explainability, which must be guaranteed before ML-based MDSS are trusted by medical practitioners. Model interpretability has become a central issue for ML in recent times, due the success of deep learnng models, paradigmatic examples of lack of interpretability. We have tried to convey the message that medical data analysts must widen their scope to ensure the interpretability of the complete analytical process by involving medical experts in it, with special attention paid to interpretability achieved by interactive visualization. We should ensure that the interaction between the data analysts and the medical experts adheres to a formal protocol in which the specific requirements of each of these parties are clearly and unambiguously laid out. This form of interactive ML makes methodically correct analytical processes more difficult to implement, evaluate and replicate. These difficulties, though, should be offset by the advantages of adhering to such protocol, which would maximize the chances of ML-based MDSS being integrated in the routine of clinical practice.

ACKNOWLEDGMENT

This is an abbreviated version of [5] Vellido, A. (2020). doi.org/10.1007/s00521-019-04051-w. The study is funded by Spanish research project PID2019-104551RB-I00.

REFERENCES

[1] Cabitza, F., Rasoini, R. and Gensini, G.F. “Unintended consequences of machine learning in medicine.” JAMA-J. Am. Med. Assoc. Vol. 318(6), pp. 517–518 (2017)
[2] Vellido, A., “Societal issues concerning the application of artificial intelligence in medicine.” Kidney Diseases Vol. 5, pp. 23–27 (2019)
[3] Gerlings, J., Jensen, M.S. and Shollo, A., “Explainable AI, but explainable to whom?.” ArXiv preprint arXiv:2106.05568 (2021)
[4] Sacha, D., *et al.* “What you see is what you can change: Human-centered machine learning by interactive visualization.” Neurocomputing Vol. 268, pp. 164–175 (2017)
[5] Vellido, A., “The importance of interpretability and visualization in machine learning for applications in medicine and health care.” Neural. Comput. & Applic. Vol. 32, pp. 18069–18083 (2020)

Signaleeg: A practical tool for EEG signal data mining

Joaquim Massana
eXiT research group
University of Girona
Girona, Spain
joaquim.massana@udg.edu

Òscar Raya
eXiT research group
University of Girona
Girona, Spain
oscar.raya@udg.edu

Jaume Gauchola
eXiT research group
University of Girona
Girona, Spain
jaume.gauchola@udg.edu

Beatriz López
eXiT research group
University of Girona
Girona, Spain
beatriz.lopez@udg.edu

***Abstract*—Due to the proliferation of brain and neurological disorders, electroencephalography (EEG) [1] is gaining attention as a support for decision making in the fields of neurology, psychology, and psychiatry. But EEG data are not always easy to understand. Therefore, extracting the desired information from EEG data in different contexts is an important requirement. This article analyses state-of-the-art EEG signal processing tools and proposes a new one: Signaleeg. This addresses the limitations of previous tools. It has been designed with the aim of helping users to build predictive models from EEG signals in a process that is called signal-data mining (DM). Moreover, Signaleeg is user friendly and multi-threaded, with optimisation facilities for finding the best predictive model. It has been implemented and tested in three scenarios: schizophrenia diagnosis, alcoholism detection, and emotion recognition. The tool provided good results in each case, thus demonstrating its versatility.**

***Index Terms*—EEG, Schizophrenia, Alcoholism, Emotions, Toolbox, Signal characterization**

Article reference This article is a summary of the original article Signaleeg [2].

I. Introduction

EEG data have been used for a lot of purposes, e.g., to desing a brain-to-computer interface (BCI), to study changes in the electrical activity of the brain as a response to an external stimulus or an internal mental process (event related potential, ERP), and to predict a brain-related health condition (prognosis). This article focuses on the use of EEG data from, e.g., the prediction of paroxysmal conditions, performing early detection of conditions, and general condition detection.

Predictive tasks require the generation of a model from the EEG data by following a data mining methodology which is denoted in the present article as signal-DM. Although there are a lot of tools that can work with EEG, these present some weaknesses, e.g., they are not user friendly, they do not offer optimization tools nor multi-threading, and they do not give examples demonstrating that the software can generate consistent results using real settings in real contexts.

This article presents Signaleeg, which is designed with the aim to overcome the limitations of current EEG software following signal-DM methodology.

Finally, we demonstrate the versatility of Signaleeg providing three use cases (schizophrenia diagnosis, alcoholism diagnosis, and emotion detection).

II. Related work

At the time of writing the article, a total of 18 state-of-the art tools used for EEG data mining are analysed. Most of the tools provide visualisation, preprocessing, and graphical user interfaces (GUIs), and the majority are based on Matlab and are freeware. Around 20% of the tools include the following options: feature extraction, connectivity, modelling, and evaluation. A small number of the tools (16%) have been designed with the idea of being user-friendly. Only 10% of the tools include a use case to demonstrate that they work properly, and only 10% incorporate multi-thread strategies for the reduction of computational time.

In contrast, Signaleeg includes all the functionalities and options of the best existing tools but is unique in that it also includes an optimisation method, and it has been tested in three different use cases.

III. Methodology

Signaleeg is based on a modular architecture that follows the signal-DM methodology. Its modules are grouped in four main blocks: utilities, preprocessing, modelling, and evaluation. All the modules can be used from the GUI which provides the user a guided methodology and explanations on what's going on. Signaleeg modules can be also used from the command console, thus, any researcher can use the desired ones as a part of their own methodology. Also, the use of Signaleeg via code enables the processing of huge amounts of data and the testing of several models without having to expend too much effort on coding.

A. Utilities: Data reading and visualisation

Several functionalities are catalogued and grouped as utilities. These are meant to be used by other modules or to improve the tool. The most important are the data reading utilities that allow the user to read different file formats. There are also utilities to visualise the data in the GUI.

B. Preprocessing

There are seven modules containing different methods for preprocessing raw data: noise reduction, data tagging, data framing, feature extraction, feature combination, data partitioning, and data balancing.

The most prominent modules are noise reduction and feature extraction. The noise reduction module allows the user to

apply band filters and remove artifacts in order to clean the raw EEG. The feature extraction module allows the user to extract a total of 30 characteristics from the signal, which will be used for modelling. Some of the features that can be extracted are related to entropy or fractal dimensions.

C. Modelling

Signaleeg's modelling module provides methods for feature selection, for generating predictive models, and for optimising the hyperparameters of the models. Signaleeg offers several methods for modelling that can be configured through GUI. The methods included are multiple linear regression (MLR), artificial neural network (ANN), classification trees (CT) and regression trees (RT), and super vector machine (SVM) and super vector regression (SVR).

D. Evaluation

Signaleeg provides several regression or classification scores. Classification scores include confusion matrix and some of its derivations. Regression scores include correlation coefficient (CC), root mean square error (RMSE), normalised root mean square error (NRMSE), mean absolute percentage error (MAPE), mean absolute error (MAE), and so on.

IV. Use cases

In this section, Signaleeg is tested in a straightforward way with three EEG data sets from a variety of application domains, to demonstrate that Signaleeg is a generic tool for signal-DM. The three prediction problems addressed are schizophrenia, alcoholism detection, and emotion recognition. Two of them concern classification and the other regression.

Similar preprocecing is applied to the tree cases extracting all features from all the channels. Then, using Signaleeg's optimization module, several models are tested and evaluated against different sets of features and channels. The scores for each model are calculated using 10 partitions with 30% of test data for cross-validation. In case of emotion recognition the outputs were binarized before and after modelling so we could be able to apply both, classification and regression methods.

The best model found for schizophrenia, with 84.16% accuracy, is a SVM with radial basis function (RBF) kernel that uses channels c5, c7, and c10, and mean delta band, mean theta band, approximate entropy, sample entropy, conditional entropy, weighted phase lag index, and Katz fractal dimension as features.

The best model found for alcoholism detection, with a 70.45% accuracy, is a CT with adaBoost kernel that uses channels c20, c50, c56, c60, and c61, and standard deviation, sample entropy, Tsallis entropy, weighted phase lag index, and generalised Hurst exponent as features.

The best model found for emotion detection, with a 64.76% accuracy, is a RT with bag kernel, that uses channel c5 and mean, approximate entropy, sample entropy, generalised Hurst exponent, and Higuchi fractal dimension as features.

V. Discussion

The Signaleeg tool has been tested using three different use cases: schizophrenia, alcoholism, and emotions. This shows that it can produce adequate results in different areas, making it adaptable and versatile.

The outcomes achieved in each data set can be compared with the results reported in the literature. For schizophrenia, [3] reported an accuracy of 85.3%, while Signaleeg achieved 84.16% accuracy. On alcoholism, [4] attained 80.83% accuracy, while Signaleeg achieved 70.45% accuracy. Finally, on emotion recognition, Signaleeg achieved 64.54% accuracy, while [5] attained 67.89%.

With regard to Signaleeg's efficiency, the computational time for training 3,800 models has been tested, needing 24 hours to train when using one thread and only 8 hours when using three threads. Therefore, the computational time can be significantly reduced using the multi-thread strategy.

In summary, Signaleeg provides a data mining methodology that can be followed in an understandable way.

VI. Conclusions

EEG data management is becoming important because of the increase in mental and neurological diseases, since it allows researchers to understand how the brain works. Of the several types of EEG praxis, this article focuses on its usage in prediction.

Signaleeg has been designed, implemented, and tested to that end, and after reviewing the state-of-art regarding tools for EEG management. It overcomes the shortcomings of existing systems by providing a user friendly GUI with a clear methodology, model optimisation, and time reduction due multi-thread strategies.

Signaleeg has been tested in three different use cases providing results close to the state-of-art in a quick and easy way, without expert knowledge.

For future work, deep learning methods will be added to Signaleeg. In fact, the modular architecture of the tool facilitates the addition of new methods.

References

[1] K. Blinowska and P. Durka, *Electroencephalography (EEG)*. American Cancer Society, 2006. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/9780471740360.ebs0418

[2] J. Massana, Ò. Raya, J. Gauchola, and B. López, "Signaleeg," *Neuroinformatics*, Jan 2021. [Online]. Available: https://doi.org/10.1007/s12021-020-09507-2

[3] A. Piryatinska, B. Darkhovsky, and A. Kaplan, "Binary classification of multichannel-eeg records based on the ϵ-complexity of continuous vector functions," *Computer Methods and Programs in Biomedicine*, vol. 152, pp. 131 – 139, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0169260716309877

[4] N. T. Rachman, H. Tjandrasa, and C. Fatichah, "Alcoholism classification based on eeg data using independent component analysis (ica), wavelet de-noising and probabilistic neural network (pnn)," in *2016 International Seminar on Intelligent Technology and Its Applications (ISITIA)*, July 2016, pp. 17–20.

[5] J. Chen, B. Hu, P. Moore, X. Zhang, and X. Ma, "Electroencephalogram-based emotion assessment system using ontology and data mining techniques," *Applied Soft Computing*, vol. 30, pp. 663 – 674, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1568494615000083

Radiomics textural features extracted from subcortical structures of grey matter probability for Alzheimers disease detection.*

César A. Ortiz Toro*, Nuria Gutiérrez Sánchez†, Consuelo Gonzalo-Martín‡, Roberto Garrido García§, Alejandro Rodriguez González¶ and Ernestina Menasalvas Ruiz‖
Centre for Biomedical Technology
Universidad Politécnica de Madrid
Pozuelo de Alarcón, Spain
Email:*ca.ortiz@upm.es, †nuria.gutierrezs@alumnos.upm.es, ‡consuelo.gonzalo@upm.es,
§roberto.garrido.garcia@alumnos.upm.es, ¶alejandro.rg@upm.es, ‖ernestina.menasalvas@upm.es

***Abstract*—Alzheimer's disease (AD) is characterized by a progressive deterioration of cognitive and behavioural functions as a result of the atrophy of specific regions of the brain. Its severity, prevalence and the long period in which the disease remains asymptomatic make it urgent to find biological markers for early detection and monitoring. In this work, (part of the 2019 IEEE 32nd International Symposium on Computer-Based Medical Systems (CBMS)) it is shown an analysis of textural radiomics features extracted from a gray matter probability volume, in a set of individual subcortical regions, from a number of different atlases, to identify subject with AD in a MRI. Also, significant subcortical regions for AD detection have been identified using a ReliefF rank test. Experimental results using the ADNI1 dataset have proven the potential of some of the tested features as possible biomarkers for AD/CN differentiation.**

***Index Terms*—Alzheimer's disease, Radiomics, Support vector machines, Magnetic resonance imaging**

I. Introduction

To this day, the diagnosis of Alzheimer's disease (AD) remains essentially clinical, meaning that it is generally not detected until the first symptoms appear, when the neuropathological damage is already significant. It is estimated that the evolution of this disease takes between 20 and 30 years from its pre-clinical stage, until recognizable and conclusive symptoms are presented for the diagnosis of AD [1]. Therefore, detection and monitoring of AD from possible early stage biomarkers has become a necessity.

Among the set of possible biomarkers for AD, those based on the analysis of the different modalities of medical imaging, especially MRI, have demonstrated considerable potential as a method to directly measure tissue degeneration associated with AD [2]. Among these biomarkers, the use of radiomic textural features [3] have proved its value in the detection of AD [4]. The concept of radiomics involves the conversion of digital medical images into high-dimensional minable data, using a large panel of phenotypic features to unveil information associated with possible underlying pathophysiological anomalies. So far, radiomic studies of AD have focused on extracting features from the MRI volume intensity values. However, the inherent variability of brain tissues [5] and the differences derived from the model and configuration of the MRI hardware can significantly affect those textural analyses [6].

As the neuropathological damage in AD progresses in a known pattern, it is possible to identify the disease in its early stages by focusing on specific Regions of Interest (ROIs) of the brain. The generation of biomarkers based on specific brain ROIs relies on the partitioning of the original MRI volumes in a set of cortical regions using an existing atlas template. However, despite the existence of multiple atlases of the brain, there is no accepted standard for the partitioning of the cortex and subcortical structures, or for the assignment of labels to the resulting ROIs [7] which represents a considerable problem when it comes to selecting ROIs in the study of discriminatory features extracted from specific brain regions.

In this study, we propose the characterisation of a set of subcortical areas using individual radiomic features in order to generate models for the detection of AD in MRI images. Unlike previous studies, a volume representing the probability that each voxel of a MRI volume corresponds to grey matter is used as a basis for feature extraction. We also compare the performance achieved using different atlases. In addition, to identify which ROIs in relation to which atlases and features are relevant for AD detection, significance tests are performed.

II. Materials and Methods

A three-stage analysis is proposed. In the first stage, the MRIs are processed in order to carry out bias correction and spatial normalization of the data set, using the Clinica software platform (version 0.1.0)[1]. This platform also provides an analysis of the MRIs volumes, representing the probability of being white matter, grey matter and cerebrospinal fluid. In the second stage, from the volume of grey matter, textural features are extracted from the ROIs. Finally, a classification

This paper is supported by European Unions Horizon 2020 research and innovation programme under grant agreement No. 727658, project IASIS.

[1]www.clinica.run/www.aramislab.fr

TABLE I
CLASSIFICATION RESULTS FOR THE DIFFERENT ATLAS AND RADIOMIC TEXTURAL FEATURES; F1-SCORES

	AAL2	Hammers	LPBA40	Neuro.	AICHA
Mean (Clinica)	0.741	0.750	0.762	0.839	0.783
Autocorrelation	0.814	0.785	0.872	0.867	0.851
Cluster T.	0.641	0.692	0.792	0.730	0.727
Correlation	0.378	0.654	0.679	0.590	0.558
Sum Average	0.807	0.800	0.851	0.823	0.807
Sum Entropy	0.487	0.692	0.821	0.800	0.814
Sum Squares	0.690	0.720	0.807	0.730	0.771
SALGLE	0.521	0.518	0.595	0.450	0.641
SAHGLE	0.510	0.701	0.763	0.565	0.769
LALGLE	0.432	0.444	0.487	0.500	0.428
LAHGLE	0.909	0.821	0.877	0.928	0.872

model is trained for each radiomic feature and atlas. In order to assess the relevance of the features related to each ROI, the weights provided by the ReliefF rank importance algorithm [8] is used.

A set of 72 subjects, 44 Cognitive Normal patients (CN) and 28 AD, are included in this study, obtained from the Alzheimer's Disease Neuroimaging Initiative (ADNI).

Five atlases (AAL2, Hammers LPBA40and Neuromorphometrics and AICHA) are tested. 9 subcortical regions (right/left hemispheres) relevant to AD progression [9], are selected as ROIs: hippocampus, parahippocampal gyri, amygdala, middle temporal gyri, superior temporal gyri, lateral orbital gyri, medial orbital gyri, cingulate gyri and precuneus.

Taking into account the particularities of the volume of grey matter probability, 10 textural features have been tested. From the grey level co-occurrence matrix, for each region of interest, autocorrelation, cluster tendency, correlation, sum average, sum entropy and sum of squares or variance are retrieved. Also, specifically for the evaluation of the homogeneity in the distribution of intensities, using the grey-level size zone matrix, Small Area Low Grey Level Emphasis (SALGLE), Small Area High Grey Level Emphasis (SAHGLE), Large Area Low Grey Level Emphasis (LALGLE) and Large Area High Grey Level Emphasis (LAHGLE) [10].

To perform the categorization of MRI volumes, models generated using Support Vector Machine (linear kernel, cost parameter = 1) are used. The models are trained and examined using ten-fold cross-validation. To compare, we use as a baseline the results obtained with a model generated from the mean intensity of each region of the complete parcelation of the brain, for each atlas, provided by Clinica.

III. RESULTS AND CONCLUSIONS

The results of the classification experiments are summarized in Table I. Among the set of tested features, the features LAHGLE (for atlases AAL2, Hammers, LPBA40 and Neuromorphometrics), autocorrelation and sum averages (for atlases AAL2, Hammers and Neuromorphometrics) show significant improvements for most of the proposed evaluation metrics (McNemar test $p < 0.05$), compared to the baseline classification model. Although these features also show improvements in relation to the baseline model in the rest of the atlases, but it is not possible to claim statistical significance for these results. On the other hand, features as SALGLE, SAHGLE, LALGLE and correlation result in significantly worse classifiers models than those obtained with the baseline features (McNemar test $p < 0.05$). The rest of the features (sum entropy, sum squares and cluster tendency) generated models with no significantly improvements over its baseline models.

Tested textural features show consistent behaviour between atlases; features that produce good results in one atlas produce good results in others, and vice versa.

Among the ROIs, the hippocampus and amygdala are the most dominant for most of the radiomic features, usually accounting for between a quarter and a third of the weights obtained in the ReliefF test. This result is consistent with previous studies [4]. The temporal, middle and parahippocampal gyri are also noteworthy, especially in the case of the features of correlation, sum of averages, and LAHGLE, although there are differences between the atlases. This suggests that there are substantial differences in how these ROIs have been defined.

The results presented in this work show the potential of some of the tested radiomic features (i.e LAHGLE) as possible biomarkers in the detection of AD. As in previous studies, it was found that the hippocampus and the amygdala are the most dominant regions for MRI AD/CN differentiation. This works suggests that, taking into account only the overall performance measures for the different features, both Neuromorphometrics and Hammers have a slight advantage over the rest as the best suited atlases for the Alzheimer's disease detection, although performance differences are, in general, small.

REFERENCES

[1] B. Dubois, H. Hampel, H. H. Feldman, P. Scheltens, P. Aisen, S. Andrieu, H. Bakardjian, H. Benali, L. Bertram, K. Blennow *et al.*, "Preclinical alzheimer's disease: definition, natural history, and diagnostic criteria," *Alzheimer's & Dementia*, vol. 12, no. 3, pp. 292–323, 2016.

[2] S. Rathore, M. Habes, M. A. Iftikhar, A. Shacklett, and C. Davatzikos, "A review on neuroimaging-based classification studies and associated feature extraction methods for alzheimer's disease and its prodromal stages," *NeuroImage*, vol. 155, pp. 530–548, 2017.

[3] V. Kumar, Y. Gu, S. Basu, A. Berglund, S. A. Eschrich, M. B. Schabath, K. Forster, H. J. Aerts, A. Dekker, D. Fenstermacher *et al.*, "Radiomics: the process and the challenges," *Magnetic resonance imaging*, vol. 30, no. 9, pp. 1234–1248, 2012.

[4] A. Chaddad, C. Desrosiers, and T. Niazi, "Deep radiomic analysis of mri related to alzheimers disease," *IEEE Access*, vol. 6, pp. 58 213–58 221, 2018.

[5] F. Kruggel, J. S. Paul, and H.-J. Gertz, "Texture-based segmentation of diffuse lesions of the brains white matter," *NeuroImage*, vol. 39, no. 3, pp. 987–996, 2008.

[6] L. R. Schad, "Problems in texture analysis with magnetic resonance imaging," *Dialogues in clinical neuroscience*, vol. 6, no. 2, p. 235, 2004.

[7] A. C. Evans, A. L. Janke, D. L. Collins, and S. Baillet, "Brain templates and atlases," *Neuroimage*, vol. 62, no. 2, pp. 911–922, 2012.

[8] I. Kononenko, E. Šimec, and M. Robnik-Šikonja, "Overcoming the myopia of inductive learning algorithms with relieff," *Applied Intelligence*, vol. 7, no. 1, pp. 39–55, 1997.

[9] K. A. Johnson, N. C. Fox, R. A. Sperling, and W. E. Klunk, "Brain imaging in alzheimer disease," *Cold Spring Harbor perspectives in medicine*, p. a006213, 2012.

[10] R. M. Haralick, K. Shanmugam *et al.*, "Textural features for image classification," *IEEE Transactions on systems, man, and cybernetics*, no. 6, pp. 610–621, 1973.

Automatic pneumonia detection in chest X-ray images using textural features

César Ortiz-Toro[***],Angel García-Pedrero[‖**†],Mario Lillo-Saavedra[††‡],Consuelo Gonzalo-Martín[‖**§]
[‖]*Centre for Biomedical Technology*,[**]*Universidad Politécnica de Madrid*
Pozuelo de Alarcón, Spain
[††]*Faculty of Agricultural Engineering, University of Concepción*
Chillán, Chile
Email:*ca.ortiz@upm.es, †angelmario.garcia@upm.es, ‡malillo@udec.cl, §consuelo.gonzalo@upm.es

***Abstract*—Fast and accurate diagnosis is essential for the triage and management of pneumonia, particularly in the current scenario of a COVID-19 epidemic outbreak, where this pathology is a major symptom. With the objective of providing tools for that purpose, this work (currently under review for publication in PLOS ONE) assesses the potential of three textural image characterisation methods: fractal dimension, radiomics and the recently developed superpixel-based histon, as biomarkers to be used for training Artificial Intelligence models in order to detect pneumonia in chest X-ray images. Two different open access image datasets are used in this study. On the first one, a dataset composed of paediatric images, the best generated models achieve a 83.3% accuracy for radiomics, 89.9% accuracy for fractal dimension and 91.3% accuracy for superpixels based histon. On the second one, a dataset derived from an COVID-19 image repository, the best generate models result in a 95.3% accuracy for radiomics, 99% accuracy for fractal dimension and 99% accuracy for superpixels based histon. These results support the effectiveness of the tested approaches.**

***Index Terms*—Pneumonia; X-ray; fractal dimension; radiomics; histon; superpixels; Diagnostic imaging; COVID-19;**

I. Introduction

Pneumonia is a form of acute respiratory infection, typically caused by viruses or bacteria. Although it affects people of all ages and is usually a mild disease, it is one of the major infectious causes of death among vulnerable groups, such as the elderly and especially children. Thus, in 2017, this disease was associated to the deaths of over 808,000 children under the age of five, worldwide, accounting for 15% of all deaths in this age group. Early detection and treatment of pneumonia is essential [1]. Unfortunately, the time available to physicians to perform this analysis is limited. This has been exacerbated by the conditions imposed by the recent COVID-19 pandemic. Therefore, the development of tools to support the diagnosis of pneumonia, especially when based on a common imaging modality such as X-rays, represents an interesting opportunity for the application of artificial intelligence techniques.

Therefore in the context of a COVID-19 pandemic whose one of its major symptoms is pneumonia, we further explore the possibilities provided by the textural characterization of chest X-ray images aimed at the detection of this disease, in order to improve prognostic predictions for triage and patient care management. Typically, textural image characterization offers good performance with very low computational complexity, and since it is usually based on well known techniques, its implementation is usually quite straightforward [2].

For this purpose, three substantially different methods of texture characterization have been selected: fractal dimension, super-pixel-based histon and radiomics. Fractal dimension [3] represents a well known descriptor, which has been used in many medical image classification works. Superpixel-based histon is a novel image descriptor based on the image segmentation work proposed in [4], developed in its current form as imaging biomarker in [5]. Finally, we will evaluate the use of a set of classic image descriptors grouped under the general term of radiomics [6]; since there are multiple prior works where this set of descriptors is used in a complete or partial way for pneumonia detection in X-ray images (although not in the specific configuration proposed in this work) it will serve both as reference and as assessment of previous studies.

II. Materials and methods

Two different datasets, covering two different fields of application, are used in this study. The first one, (referred here as GWCMCx), is a paediatric dataset from the Guangzhou Women and Children Medical Center (Guangzhou, China) composed of 5840 X-rays images, divided into a training set consisting of 5216 X-rays and a testing set of images of 624 images. The second dataset (named here as Josep-NIH) combines images from two datasets, the COVID-19 collection [7] and the National Institutes of Health Clinical Center (NIH) dataset [8]. These images have been randomly divided into a training and a test set of 1156 and 300 images respectively. Taking into account the differences in both the demographics of the subjects covered and the overall appearance and quality of the images, separate models will be trained for each dataset.

The GWCMCx dataset is not a balanced dataset, as it contains almost three times more pathology-free images than images with pneumonia. As a result, some Machine Learning (ML) algorithms may result in models biased towards the majority class. In this work, we compare the differences between an unbalanced dataset and a balanced dataset created using the Synthetic Minority Oversampling Technique (SMOTE) [9]. Before generating the models, images are processed for intensity normalization, masking of obvious non-lung areas and image texture enhancement.

To prevent potential biases arising from the ML technique applied, three different ML techniques have been tested: k-Nearest Neighbours (KNN), Support Vector Machine (SVM) and Random Forests (RF). We compare the results obtained both with the default configuration ($K = \sqrt{N}$ where N is the training set size for KNN; linear kernel and the cost parameter =1 for SVM; 100 learning cycles and 2 leaf per node, for RF) for these algorithms and with the parameters obtained after a process of hyperparameter optimization. To generate each classification model, we apply a 10-fold cross-validation methodology using the training sets. The best model, measured by the achieved F-Scores, is evaluated using the tests sets. The goal is to test whether the model will be able to generalise over an independent dataset while avoiding problems such as overfitting or selection bias [10]. Experiments have been carried out using MATLAB 2020a.

III. Results and Conclusions

The best results for the unbalanced GWCMCx dataset (F-Score 91%-92%, McNemar test p<0.05, compared to the rest of the approaches) are achieved by models generated from a histon-based superpixel characterisation using KNN, (McNemar p<0.05). For the remaining generated models, the differences in performance are minor and can only be considered significantly for the models generated for radiomics with KNN and SMV, the poorest performance results. Errors tend to be false positives. This is to be expected considering the training set bias towards non pneumonia images.

Similarly to the previous case, the best result (F-Score 93%) for the GWCMCx SMOTE balanced dataset is achieved by a model trained from a superpixel-based histon characterization using KNN (McNemar's test $p < 0.05$). The rest of the models trained from both superpixel-based histon and fractal dimension achieve equivalent performance; the minor differences between them cannot be considered statistically significant (McNemar p¡0.05). The poorest results are again obtained using radiomics, although in this case, when compared with the rest of the models, the differences are relevant and statistically significant (McNemar p<0.05). For this models, the differences between false positives and negatives are minor, with the miss-classifications types moving towards equilibrium.

Globally, the results obtained using both the balanced GWCMCx test set with radiomics and the histon characterisations represent an improvement over their counterparts in the unbalanced GWCMCx test set (McNemar p<0.05). Although there is an increase in the number of false negatives, this is compensated by a decrease in false positives, resulting in a better overall performance. On the other hand, in the case of the models generated with radiomics, the increase in true positives does not compensate for the increase in false positives, with the performance of the generated models being worse. The exception to this case corresponds to the models generated with RF, where there are no relevant statistical differences in the performance of the models generated with the balanced and unbalanced ensemble (McNemar p<0.05).

The performance results show the impact of using training sets where the classes are not balanced. Overall, there is an improvement in performance results when using a balanced GWCMCx dataset for training. Moreover, the use of balanced datasets also affects the behaviour of the obtained errors; the error bias presented in models trained with an unbalanced dataset disappears when using a balanced set.

For the Josep-NIH dataset, the best results (F-Score 98%-99%) are associated with the models generated using superpixel-based histon through the RF algorithm, and with those generated with fractal dimension as textural feature, with any classifier and hyperparameter optimisation. The differences in the performance measures between the models mentioned cannot be considered statistically significant (McNemar p<0.05). The weakest results, although still remarkable, are once again achieved through the use of radiomics.

Taking into account all the models generated, hyperparameter optimisation has been found to have a minimal or, in some cases, negative effect on model performance.

The models evaluated confirm the validity of the proposed methods. While all characterisation methods achieve remarkable results, those obtained with both superpixel-based histon and, to a lesser extent, fractal dimension are particularly noteworthy. Compared to the more classical approach based on radiomics features, the best models trained with these approaches result in improvements between 6% and 8% both for accuracy and F-Score, for the first dataset. For the second dataset, accuracy improvements are between 4% and 5%, and around a 4% for the F-Score.

References

[1] K. Kallander, D. H. Burgess, and S. A. Qazi, "Early identification and treatment of pneumonia: a call to action," *The Lancet. Global Health*, vol. 4, no. 1, p. e12, 2016.

[2] A. Humeau-Heurtier, "Texture feature extraction methods: A survey," *IEEE Access*, vol. 7, pp. 8975–9000, 2019.

[3] K. Foroutan-pour, P. Dutilleul, and D. L. Smith, "Advances in the implementation of the box-counting method of fractal dimension estimation," *Applied mathematics and computation*, vol. 105, no. 2-3, pp. 195–210, 1999.

[4] M. M. Mushrif and A. K. Ray, "Color image segmentation: Rough-set theoretic approach," *Pattern Recognition Letters*, vol. 29, no. 4, pp. 483 – 493, 2008.

[5] C. A. O. Toro, C. Gonzalo-Martín, A. García-Pedrero, and E. Menasalvas Ruiz, "Supervoxels-based histon as a new alzheimers disease imaging biomarker," *Sensors*, vol. 18, no. 6, p. 1752, 2018.

[6] R. J. Gillies, P. E. Kinahan, and H. Hricak, "Radiomics: images are more than pictures, they are data," *Radiology*, vol. 278, no. 2, pp. 563–577, 2016.

[7] J. P. Cohen, P. Morrison, L. Dao, K. Roth, T. Q. Duong, and M. Ghassemi, "Covid-19 image data collection: Prospective predictions are the future," 2020.

[8] X. Wang, Y. Peng, L. Lu, Z. Lu, M. Bagheri, and R. M. Summers, "Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2097–2106.

[9] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.

[10] G. C. Cawley and N. L. Talbot, "On over-fitting in model selection and subsequent selection bias in performance evaluation," *The Journal of Machine Learning Research*, vol. 11, pp. 2079–2107, 2010.

NIO: Neurocognitive Integrated Ontology

1st Alba Gómez-Valadés
Department of Artificial Intelligence
UNED
Madrid, Spain
albagvb@dia.uned.es

2nd Rafael Martínez Tomás
dept. name of organization (of Aff.)
UNED
Madrid, Spain

3rd Mariano Rincón
Department of Artificial Intelligence
UNED)
Madrid, Spain

***Abstract*—Early detection of mild cognitive impairment has become a priority in Alzheimer's disease research. However, the data is scattered at different domains and under various formats and standards, which makes work difficult for both humans and machines. In this context, this study represents a key note of an already published paper in which we described the development of an ontology that integrates and homogenizes all the knowledge acquired over time has become crucial. In this key work we present an article that introduces a first approach to that ontology, the Neurocognitive Integrated Ontology (NIO), which integrates the knowledge of the four main domains related to mild cognitive impairment: neurodegenerative diseases, diagnostic tests, cognitive functions and brain areas. This ontology enables interoperability and integrates the knowledge dispersed across different disciplines. To ensure the stability and reusability of the ontology, it was developed following the ontology-building life cycle.**

***Index Terms*—ontology development; MCI; Alzheimer's disease; neurodegenerative disease; Mild Cognitive Impairment**

I. Introduction

Mild Cognitive Impairment (MCI) has attracted great attention in the last decades since it is a transition phase between normal decline and dementia, like Alzheimer's disease (AD). Various tests have been developed for the early detection of MCI, i.e., biomarker tests, brain imaging tests, and neuropsychological tests (NT). Of these, NTs have several advantages, such as that they are faster, cheaper and do not need specialized equipment. However, they also present problems, such as being highly subjective, or requiring too much time when it is necessary to apply several of them because they are totally independent despite having overlapping components. The latter makes the tests redundant or of low relevance and all the information they provide is not really used. In this context, one of the lines of research of our group is to help to reduce these problems by introducing structure and machine learning techniques to obtain more efficient, objective, and experience-based tests that could help to find cognitive problems in its earlier stages in both graphic and oral production tests [1], [2]. Since the data is scattered in different domains, under different formats, and generated by different technologies, it is difficult to interpret and use both by people and by automated systems [3], [4]. Creating the adequate infrastructure to standardize and making the information accessible is a key objective for success of research [5], and the development of ontologies has been established as one of the most appropriate solutions in the biomedical domain because the formal definition of classes, properties and relationships allows homogenization and consensus in the representation of an area of knowledge. This eases the recovery, integration and exchange of heterogeneous data from different sources for both research and clinical objectives. This way, this study depict a key note in an already published paper in [6], where we propose the development of the Neurocognitive Integrative Ontology (NIO), an ontology that integrates the knowledge of the 4 main domains that relate to the early detection of MCI: brain areas, cognitive functions, NT and AD. The underlying theory that relates them is that: (1) physical damage in different areas of the brain is correlated with different alterations in the cognitive functions of patients; (2) these alterations are detected by NT, as they are reflected as different types of failures (signs) when performing each test; and (3) failures are related to different neuropsychological and neurodegenerative diseases. Despite the great interrelationship between these domains, they are usually studied in a rather isolated way, prioritizing in each study certain areas against others. Given that by focusing on a single domain or some fragments of them, part of the information involved in the characterization of neurodegenerative diseases is ignored, an ontology that integrates the knowledge of these four domains would be of great help to researchers and clinicians involved in the investigation and early detection of MCI and neurodegenerative diseases in general.

II. Ontology Development

It is known that it is essential to reuse existing ontologies when defining a new ontology to avoid redundancy and inconsistency issues. Therefore, we, first of all, we analyzed cutting-edge ontologies related to MCI with a high probability of conversion to AD and NT, classifying them into three groups: general coverage of the AD, diagnostic test modeling, and to aid in the diagnosis of MCI. The evaluated aspects were domain coverage, upper or foundation ontologies, degree of reuse of other ontologies, internal structure, class hierarchy, and metadata. Of the reviewed ontologies, we selected the Neuropsychological Testing Ontology [7] and Alzheimer Disease Ontology [8] to cover the AD and NT domains, the Foundation Model of Anatomy [9] for the brain areas domain, and the Mental Functioning ontology [10] for the domain of cognitive functions. Next, we did a manual alignment between the selected ontologies to choose the most relevant groups of terms and removed the rest, and finally integrated the

resulting modules by importing axioms and ontology modules to create the base structure of the ontology. This base structure was inspected for previously undetected redundancies, and afterward, their stability was checked using both the Protégé reasoner HermiT1.3.8 and the online program OOPS! (Ontology Pitfall Scanner! http://oops.linkeddata.es/) looking for both inconsistencies and pitfalls. Then, new relationships, and object and property terms were included using the relations established in the literature as a guide. Finally, a new evaluation of the ontology stability and pitfalls was performed after those additions. The final ontology is composed by 3,303 classes and 229 properties, and is available via BioPortal at: https://bioportal.bioontology.org/ontologies/NIO

III. Ontology evaluation

To illustrate the usability and advantages of NIO we show two practical examples that show how new knowledge can be modeled and the ontology expanded.

A. Use case 1: Adapting the ontology to a new project

In the first use case, we show how to adapt the NIO ontology to integrate new data that belongs to a different project for which it has not been specifically designed. First, the terms are checked to see if they are already in the ontology, and then the new terms and relations are added under the appropriate classes. Finally, after all changes, stability and pitfall checks are performed.

B. Use case 2: Updating the ontology using a new ontology

One of the key points of ontologies is their capability of being extended. In this second use case, we update the NIO ontology using the knowledge modeled in another ontology, NDDO [11], that covers different knowledge areas. First, we analyze NDDO to identify the potential modules that are in the scope of NIO, extract them, and make a second and deeper analysis of those modules looking for redundancies with terms already modeled in NIO. We discard those modules that, although related, are more loosely related to the NIO scope, to avoid the problem of over-complexity of the ontology. After the integration, a final analysis checking the stability and the pitfalls is performed.

IV. Conclusions and future work

Within the line of our group to optimize neuropsychological tests, it has been presented here an ontology that constitutes the first step towards a unified ontology that integrates the knowledge relating to MCI and can be reusable for other researcher groups. As future work, we expect to complete NIO with new terms and relationships and to link the ontology to a machine learning system to create an expert system that helps clinicians in the early diagnosis of MCI.

Acknowledgment

We would like to thank Maria del Carmen Mardomingo and Sara García Herranz for the permission to use their database in our project.

References

[1] J. M. Guerrero, R. Martínez-Tomás, M. Rincón, and H. Peraita, “Diagnosis of Cognitive Impairment Compatible with Early Diagnosis of Alzheimer’s Disease. A Bayesian Network Model based on the Analysis of Oral Definitions of Semantic Categories,” Methods Inf Med, vol. 55, no. 1, pp. 42–49, 2016.

[2] A. Gomez-Valadés, R. Martínez-Tomás, and M. Rincón-Zamorano, “Ontologies for Early Detection of the Alzheimer Disease and Other Neurodegenerative Diseases,” in Understanding the Brain Function and Emotions, 2019, pp. 42–50.

[3] F. F. Costa, “Big data in biomedicine,” Drug Discovery Today, vol. 19, no. 4, pp. 433–440, Apr. 2014.

[4] J. Decety and J. Cacioppo, “Frontiers in Human Neuroscience: The Golden Triangle and Beyond,” Perspectives on Psychological Science, vol. 5, no. 6, pp. 767–771, Nov. 2010.

[5] A. Burgun and O. Bodenreider, “Accessing and Integrating Data and Knowledge for Biomedical Research,” Yearbook of Medical Informatics, vol. 17, no. 01, pp. 91–101, Aug. 2008.

[6] A. Gomez-Valades, R. Martinez-Tomas, and M. Rincon, “Integrative Base Ontology for the Research Analysis of Alzheimer’s Disease-Related Mild Cognitive Impairment,” Front. Neuroinform., vol. 15, 2021.

[7] A. P. Cox, M. Jensen, A. Ruttenberg, K. Szigeti, and A. D. Diehl, “Measuring Cognitive Functions: Hurdles in the Development of the NeuroPsychological Testing Ontology,” Editorial: CEUR Workshop Proceedings, p. 6, Jul. 2013.

[8] A. Malhotra, E. Younesi, M. Gündel, B. Müller, M. T. Heneka, and M. Hofmann-Apitius, “ADO: A disease ontology representing the domain knowledge specific to Alzheimer’s disease,” Alzheimer’s and Dementia, vol. 10, no. 2, pp. 238–246, Mar. 2014.

[9] C. Rosse and J. L. V. Mejino, “The Foundational Model of Anatomy Ontology,” in Anatomy Ontologies for Bioinformatics, vol. 6, A. Burger, D. Davidson, and R. Baldock, Eds. London: Springer London, 2008, pp. 59–117.

[10] J. Hastings, W. Ceusters, M. Jensen, K. Mulligan, and B. Smith, “Representing Mental Functioning: Ontologies for Mental Health and Disease,” presented at the Proceeedings of the Third International Conference on Biomedical Ontology, Jul. 2012.

[11] A. Kostovska, I. Tolovski, F. Maikore, the Alzheimer’s Disease Neuroimaging Initiative, L. Soldatova, and P. Panov, “Neurodegenerative Disease Data Ontology,” Discovery Science, vol. 11828, pp. 235–245, 2019.

Prediction of COVID-19 severity in Madrid region with Model-leaf trees

Miguel A. Riaño-Sánchez, Concha Bielza, Pedro Larrañaga
Departamento de Inteligencia Artificial
Universidad Politécnica de Madrid
Campus de Montegancedo, Boadilla del Monte, Spain
ma.riano@alumnos.upm.es, {mcbielza, pedro.larranaga}@fi.upm.es

***Abstract*— The global pandemic caused by the Coronavirus Disease 2019 has affected society at severe levels. Among the major problems brought by it, hospital overcrowding was one of the most dangerous. This paper presents Model-leaf trees, a new machine learning classification algorithm that combines decision trees with other classification models at its leaves to create an enhanced solution. The model will be tested using COVID-19 data from 1632 patients from hospitals belonging to Fundación Jiménez Díaz, and one of the generated models will be interpreted and presented as a viable solution that could be used in a real scenario.**

Keywords*— *Coronavirus; COVID-19; Predictive analytics; Machine learning; Supervised machine learning algorithm; Madrid; Tree-based model; Supervised classification

I. INTRODUCTION

The *Coronavirus disease* (COVID-19) [1] has severely impacted society over the past year. According to the World Health Organization Coronavirus Dashboard [2], 167492769 cases of COVID-19 have been reported and it has caused 3482907 deaths by May 26th, 2021.

In a study made by Roth et al. [3] it was discovered that one of the causes of the high mortality rate between March and May 2020 was hospital overcrowding due to lack of resources. This issue concerned hospitals at the beginning of the pandemic, and doctors voiced their complaints about it several times. For this reason, tools that could help them make decisions about which patients need admission and which ones do not are highly valued.

However, these tools must justify each suggestion, as the doctor's decision is key to the patient's life. Among the supervised machine learning algorithms, one stands out from the rest due to its simplicity, interpretability and undestandability [4]: decision trees.

Decision trees have edges and nodes that generate a structure in a tree shape: a single root node at the top is divided into several nodes. Each node might also be divided into other nodes and so on, until each node is not divided anymore. Those final nodes are called leaves. Each internal node, that is, each node that is not a leaf and thus is divided, has an internal test over an attribute and it decides which branch to follow from that node. Leaf nodes contain class labels instead of a test [5]. This recursive partition of the instance space into smaller spaces [6] to solve the whole problem makes decision trees a divide-and-conquer algorithm.

The simplicity in its structure is what makes it highly interpretable, as each of these internal nodes can be translated into an *If x then y, else z* rule, which is intuitive for humans to understand, and the path an instance follows in the tree generates a set of rules that would explain why the tree made the final decision in its classification. This characteristic fits perfectly with the requirements of tools for doctors.

Examples of decision tree based proposals that were released during the past year are [7] in which the authors predict the severity of patients, or [8] in which the researchers predict their mortality rate.

However, decision trees suffer from the *fragmentation problem*: as each split is made, the data is divided based on the test and after several levels there is not much data to make decisions [9]. One of the solutions to this problem was a hybrid decision tree algorithm called: NBTree [9]. His creator noticed that the naïve Bayes classifier algorithm could be used at the leaves to perform the final classification, solving the problem. The result was a new algorithm that is able to improve the performance of decision trees.

The addition of classification models to decision trees has not always been approached by replacing the tree leaves with them. Gama and Brazdil [10] joined a supervised classification algorithm called Cascade Generalization with decision trees, naming the result CGTree. They thought about the possibilities that divide-and-conquer strategies could bring to Cascade Generalization, as the relations that could not be captured in the data at global level might be discovered by dividing the instance space. The result was another algorithm that improved the performance of Cascade Generalization by itself.

Two questions could be risen from these two works: could other classification models improve the performance of decision trees just as naïve Bayes does in NBTree? And, could other classification models be improved by applying a divide-and-conquer strategy to them? In this study we present *Model-leaf tree* (MLTree), a new supervised machine learning algorithm that combines both ideas to elaborate an enhanced

solution. The idea is not only to improve any possible supervised classification model by dividing the whole problem into subproblems, but also to improve the decision trees themselves by solving the *fragmentation problem* presented before. We will then apply this new algorithm to COVID-19 datasets in order to generate models that will help doctors in the admission of patients.

This research paper is divided into four additional sections: Section II reviews related solutions to this new algorithm. Section III presents MLTree and includes the algorithm explanation, its characteristics and functionalities. The experiments with the COVID-19 datasets and their results are shown in Section IV. Finally, Section V contains some conclusions about this research.

II. Related Work

There have been many different proposals that were aimed at combining decision trees with other models in order to design an enhanced solution.

The NBTree [9], as mentioned above, is a fusion of decision trees and naïve Bayes classifiers that combines the advantages of both models by replacing leaf nodes with naïve Bayes classifiers.

A similar solution that uses this same idea of substituting leaf nodes with other models, in this case feedforward neural networks due to its powerful nonlinear processing ability, is called Hybrid Decision Trees (HDT) [11]. The algorithm uses the categorical features in the dataset to grow the tree and, in those cases in which a node cannot be split but the numerical features have diversity in their values, it is transformed into a feed forward neural network trained with the instances at the node.

In another proposal called Logistic Model Trees [12], authors substituted tree leaves with logistic regressors. However, instead of generating one model per leaf, they generated a group of them during the tree growth: in the root, they start training logistic regression models as long as it improves the fit to the data. Then, a split is made using the same criteria as the C4.5 algorithm. In each child they continue adding more logistic regression models, using the instances in the child node as data. This continues until the leaf, that will contain a committee of linear regression models that were trained on increasingly smaller subsets of data.

Decision trees were also used in Gil-Begue, Larrañaga and Bielza [13] to design a new algorithm called Multi-dimensional Bayesian Network Classifier Trees (MBCTrees), that has multi-dimensional Bayesian network classifiers in the leaves. It is a similar approach as NBTrees, but the authors recommend using a wrapper approach to generate them so that, when doing the split, the chosen candidate will be the one that best divides the data and the one that achieves the highest global accuracy.

As stated in the previous section, the addition of new models to decision trees has not always been approached by replacing the leaves with them. CGTree [10] incorporates a divide-and-conquer strategy to Cascade Generalization using decision trees so that relations that could not be found in each problem might be found in the subproblems.

Finally, in another research [14] on combining multiple classifiers that involves the usage of decision trees, the authors presented a new kind of decision tree algorithm called Meta Decision Trees (MDT), that are able to predict, for a dataset and a list of classification models, which algorithm should predict each sample.

III. The Algorithm

The algorithm proposed in this paper is built on top of an implementation in Python of the decision tree CART algorithm [15] created by Joachim Valente [16]. Both include only the classification functionality, and they work with numerical and Boolean features (True or False).

The CART algorithm generates a binary decision tree. It is binary because each node is divided only into two nodes. For each feature, all the midpoints between adjacent training samples with different class value are possible splitting candidates. The algorithm performs a greedy splitting search while growing the tree, so that for each splitting candidate it chooses the one that minimizes the weighted sum of the node impurities in the two child nodes that would be generated. As impurity function, CART [15] uses the generalization of the binomial variance, or Gini index. This is repeated until a stopping criteria is met.

Two stopping criteria for the growing process are implemented in [16]: a hyperparameter that determines the maximum depth that the tree can reach and, when looking for the best split at a node, the weighted impurity of the children must be lower than the impurity in the current node. These two conditions would work as an early stopping (also called pre-pruning) method to prevent the tree from overfitting [17].

As the objective of this paper is to elaborate a decision tree algorithm enhanced with other classification models in some leaves, it is necessary to modify the growing process. The final algorithm can be found in Figure 1. Some important remarks will be explained below.

A. Operating options

The innovation of this algorithm is the possibility of choosing what type of classification model (*extra model*) to put in the leaves. Thus, it offers three operating options:

1. One *extra model*: the user can choose a classification model to enhance the decision tree.
2. Voting majority in model list: the user can choose a list of classification algorithms to enhance the decision tree. This list will be interpreted as a single *extra model*, and the resulting prediction will be the majority vote of each individual classification model.
3. Best model in list: the user can choose a list of classification algorithms and, each time MLTree decides that a leaf should be turned into a model, it will choose the one that maximizes the accuracy

with the instances that reached that leaf, and it will be considered the *extra model* in that node.

B. Calculation of the best splitting candidate

The calculation of the best splitting candidate in this new algorithm is the same as in the CART algorithm with one addition taken from the NBTree study: instead of only making sure that the average of the Gini index has to be lower than the impurity on the current node, in this new approach the reduction in the impurity should be at least of a 5%.

C. Calculation of the model accuracy

This new term, *model accuracy*, in the current node is just measuring the 5-fold cross-validation accuracy of the *extra model* trained in the node with the instances it has. The *model accuracy* in each child node is, nevertheless, calculated in a different way:

- If all the instances in the child node belong to the same class, the *model accuracy* in the child is 1.
- If there are fewer than 5 instances of each class label in the child node, its *model accuracy* is calculated as the number of instances that belong to the majoritarian class label divided by the total number of instances.
- If the child node does not meet the previous two points, then the *model accuracy* is calculated as it was described above, with the 5-fold cross-validation accuracy.

After the *model accuracy* in each child node has been found, the weighted *model accuracy* in the children is calculated as the sum of the *model accuracy* of each one multiplied by the proportion of instances from the current node each of them has.

Input: a dataset D of labelled instances.

Output: a decision tree enhanced with other models in some of its leaves.

- If the node meets any of the group 1 stopping conditions, it is turned into a leaf and the predicted class is the most repeated one among the instances in the node.
- Look for the best splitting candidate C.
  - If it is found, then calculate the *model accuracy* on each child that would be generated. Compare the *model accuracy* in the current node with the weighted *model accuracy* of the children:
    - If it is better in the children, then partition D according to C in two, because it was checked that this action will have an improvement not only in the impurity of the instances but also on the accuracy.
    - Repeat for each child generated in the split with its correspondent division of D.
- If the node meets any of the group 2 stopping conditions, then it is transformed into a leaf and an extra classification model is trained with the instances in the current node.

Fig. 1. Model-leaf tree algorithm.

D. Stopping criteria

As the CART algorithm, this new approach has also some stopping conditions to improve the quality of the resultant tree by avoiding the node to be split. These conditions can be divided into two groups:

Group 1: Stopping conditions that generate a leaf

- All the instances belong to the same class label.
- In the node, there are fewer than 5 instances per class label.

As the *model accuracy* is calculated using the 5-fold cross-validation, the programming function that is being used requires at least one instance per class label in each fold, so at least there should be 5 instances per class label.

These stopping conditions are checked at the beginning of each iteration of the algorithm, before making calculations to generate the split. If any of them are met, the node is transformed into a traditional decision tree leaf, in which the predicted class label is the most repeated one among the instances in the node.

Group 2: Stopping conditions that generate an extra classification model

- The tree has reached the maximum depth.
- The best threshold to make the split was not found.
- The node has fewer than 30 instances.
- The *model accuracy* in the node is better than the weighted *model accuracy* in the children.

The first two conditions were already in the CART algorithm, as explained before. The last two, however, were taken from the NBTree [9]. The author considered that the split should not be done in the node if it has less than 30 instances. This is mainly due to the fact that the resulting nodes would have fewer instances and the possible *extra models* that could be trained might have low quality. Moreover, he also stated that the *model accuracy* in the current node should be lower than the weighted *model accuracy* in the children. This way, when a split is done, it is not only assuring that the resulting nodes will have less impurity in their instances, but also that a possible model trained with them will be better than a model trained in the current node with all the samples in it, which differences this new algorithm from other proposals like [11].

E. Class weights

Last, but not least, this new algorithm offers the user the possibility of assigning weights to each class label as an argument. Each weight must be a positive integer.

In a two-class label classification problem, the algorithm will prioritize the correct classification of the instances with the class label that has the highest weight. Consequently, instances with the other class label will be classified worse. This is achieved in this implementation by:

- Changing the way accuracies are calculated: the only accuracy that is not calculated using a *Sci-kit learn* [18] function is the one when a child node

has less than 5 instances per class label. In that case, when counting the number of instances for each class label *when performing the calculation*, each sample will count as their weight. That is, for each class label L with its weight W_L, each instance with class label L will count as W_L instead of 1.

- Changing the training of the *extra model*: the weights will be passed as arguments to the *Sci-kit learn* [18] functions that generate each *extra model*, so that they are taken into consideration.

IV. Experiments and Results

If it were possible to predict if a COVID-19 patient would evolve unfavorably or not, we could develop a tool that would help doctors in the emergency department to decide whether a patient has to be admitted or not. In this section, the *MLTree* algorithm will be used to generate and to study models capable of achieving this.

A. Dataset description and preprocessing

To get a model that is able to distinguish which COVID-19 patients will evolve favorably and thus need admission and which ones will not, medical data on patients with this disease are needed. For this paper, we are grateful to have such dataset from hospitals belonging to Fundación Jiménez Díaz. It contains data from COVID-19 positive patients that were registered in the hospital at some point for any reason, like a consultation or admission. Therefore, it needed to be preprocessed in order to keep only the most useful data to elaborate the model that is needed.

When a patient arrives at the emergency department, doctors have only their personal information, their comorbidities and the measures taken for the triage. Then, if the patient is admitted, he/she usually undergoes a blood test. At the beginning of this pandemic, testing laboratories were as overcrowded as hospitals. Because of these facts, it was planned to generate two datasets: *emergency*, that includes only the available measures when a patient arrives at the hospital to generate models that could help doctors decide whether it is worth admitting the patient and ordering a blood analysis or not, and *analysis*, with the same attributes as *emergency* plus all possible blood analysis measures. The decided list of features to be extracted from the raw dataset included: personal information (age, sex and body mass index), comorbidities (like smoker and diabetes, among others), measures taken for the triage (first oxygen saturation and first body temperature) and several measures from blood analysis (like albumin or urea). To achieve this goal, some operations were performed over the data.

First, typical transformations in data analysis were made on the raw dataset: variable rename, generating useful columns (admission time to indicate how much time the patient stayed at the hospital and death and alive, to mark which patients passed away). Missing values were imputed using an iterative imputation with linear regressions. Afterwards, three important filtering steps would generate the final datasets.

The model needs to distinguish patients whose disease will evolve unfavorably from those that will not. For this paper, a patient is considered critical (the disease will evolve unfavorably) if he/she dies within 72 hours after their last test. A patient is considered non-critical (the disease will evolve favorably) if he/she does not die before 120 hours after their last test. So, the first filter only keeps patients that were admitted in the hospital and that have had at least one blood analysis and that meet the requirements above. Then the second filter kept only the list of features described before. This step left 69 attributes, of which 56 were blood test measures, which is a high number. A meta estimator that generates 100 random decision trees was fitted using the filtered dataset. A third filter kept only the blood analysis features that were used by the majority of models. This reduced the number from 56 to 23.

In order to fit classification models, the last step was to balance the class labels (critical and non-critical). After all this preprocessing, *emergency* has 13 features and *analysis* 36, and they both have instances from 1632 patients.

B. Performance evaluation of the algorithm

In this paper both datasets, *emergency* and *analysis*, will be used to test the MLTree algorithm performance. The 5-fold cross-validation accuracy will be used as metric for the evaluation.

For this analysis, MLTree will use four models taken from the Python module *Sci-kit learn* [18]: *SVC* (a C-support vector classification algorithm), *KNN* (a k-nearest neighbors vote classifier), *LR* (a logistic regression classifier that uses the solver liblinear) and *BAG* (a bagging meta-estimator that fits 20 SVC classifiers using a 75% of samples and features). The algorithm will also be evaluated with its second and third operation option using the first three classifiers presented before as a list of models.

The accuracies resulting of this performance evaluation can be observed in Table I. Each one is followed by its standard deviation (the number after the ± sign).

C. Comparison with other models

In order to assess how well the algorithm performs, a comparison with other models will be made:

- A comparison with the implementation of the CART algorithm in the Python module *Sci-kit learn* [18]. This way it can be assessed if the addition of extra classification models on the leaves improves its performance.
- A comparison with each *extra model*. This way, it will be checked if the usage of a divide-and-conquer algorithm to separate the instances for the *extra model* gets a better result than having the model alone. This last comparison will be made only with the results obtained using the first operating option of MLTree.

To compare the performance evaluations, the 5-fold cross-validation accuracies of the CART algorithm [18] and each classifier that was used in the previous section will be used

against their correspondent result from the performance evaluation of MLTree in a hypothesis test. Due to its low Type I error, Dietterich [19] proposed the usage of the McNemar's test for this situation. Its null hypothesis is that "the two models have the same error rate". The accuracies resulting of the performance evaluation using each model alone can be observed in Table II. Each one is followed by its standard deviation (the number after the ± sign). The p-values resulting from the comparison between the CART models and the MLTree models can be found in Table III. The ones resulting from the comparison between each classification model alone and its usage in MLTree can be found in Table IV.

Using a significance level of 0.05, the null hypothesis is rejected in almost all the cases when comparing MLTree with CART. So, excepting the case when *KNN* is used with *analysis* dataset in MLTree, it can be stated that the CART algorithm generates significantly different models than MLTree, being better the results by the new algorithm according to the accuracies obtained. The only case where the null hypothesis cannot be rejected might have happened due to an overfit of the data, which caused the CART algorithm to outperform the MLTree algorithm.

The results when comparing the performance evaluation of using classification models by themselves with the usage of MLTree with the same models vary depending on the dataset. Nothing can be concluded from the *emergency* dataset, as the p-value is higher than the significance level in all the cases. In the *analysis* dataset however, the null hypothesis can be rejected in 3 out of 4 algorithms, showing that, for this dataset and using *SVC*, *KNN* or *BAG* with MLTree, the generated model will be significantly different from the one generated by each algorithm itself and, according to the accuracies obtained, it would outperform it. One of the causes that prevented us from rejecting the null hypothesis could be that MLTree may have overfitted the data, as it might have also happened in one of the comparisons with the CART algorithm. However, as it can be seen in Table IV, when using the dataset *emergency* with *SVC* and when using each of the datasets with *LR*, the obtained p-value is 1. This means that MLTree actually decided that applying the divide-and-conquer strategy in those cases would not generate an improvement in the model accuracy, but that the *extra model* by itself already has the highest. MLTree is never forced to generate a decision tree in all cases, but it chooses the design that maximizes the accuracies obtained with the other classification algorithms.

TABLE I. MLTREE PERFORMANCE EVALUATION

Dataset name	Option 1				Option 2	Option 3
	Extra model: SVC	***Extra model: KNN***	***Extra model: LR***	***Extra model: BAG***		
emergency	0.6305 ±0.024	0.5668 ±0.022	0.6324 ±0.031	0.6373 ±0.025	0.6318 ±0.025	0.6207 ±0.022
analysis	0.7488 ±0.023	0.7341 ±0.012	0.7855 ±0.013	0.7426 ±0.023	0.75 ±0.025	0.7868 ±0.021

TABLE II. PERFORMANCE EVALUATION USING EACH MODEL ALONE

Dataset name	CART	SVC	KNN	LR	BAG
emergency	0.5251 ±0.046	0.6311 ±0.028	0.5692 ±0.026	0.6318 ±0.031	0.6372 ±0.028
analysis	0.7193 ±0.029	0.7028 ±0.03	0.6581 ±0.03	0.7855 ±0.021	0.685 ±0.041

TABLE III. MCNEMAR HYPOTHESIS TEST. COMPARISON WITH THE CART ALGORITHM

Dataset name	Extra model: SVC	Extra model: KNN	Extra model: LR	Extra model: BAG	Option 2	Option 3
emergency	3.7e-13	4.1e-3	3.7e-14	4.1e-13	1.4e-15	1.5e-10
analysis	9.1e-4	0.2674	8.2e-19	0.002	2e-4	4.6e-11

TABLE IV. MCNEMAR HYPOTHESIS TEST. COMPARISON WITH EACH ALGORITHM ALONE

Dataset name	SVC	KNN	LR	BAG
emergency	1	0.221	1	0.791
analysis	1.6e-4	2.94e-8	1	1.38e-4

D. Interpretation of a MLTree model

The ultimate goal of the design of this algorithm is to find a better solution that can help doctors make decisions about COVID-19 patients that arrive to the emergency department. We will make an interpretation of a MLTree model trained with logistic regression as *extra model* and using the dataset *emergency* to study if it could be used in this situation. Its visual representation is found in Figure 2.

If the features the model has chosen to divide the dataset are studied, it can be assessed that they match reality. This model is saying that patients that are below 54 years old (feature *Age*) and that have an oncologic disease (variable *Onco*) can be classified as *critical* (class label *True*) and they should be admitted, or at least it is worthwhile for the doctor to order a blood test. Patients that are in the same age range and do not have any oncological diseases are classified using a *LR* model, whose feature coefficients can also be interpreted (a higher coefficient would mean that that feature collaborates in increasing the probability for a patient to be *critical*). This first logistic regression model has assigned high positive coefficients to two comorbidities: Smoker and Diabetes.

Going down the other branch from the root, if the patient is 54 years old or more, then another *LR* model will classify the instances. In this case, it has given high coefficients to the variables *Cardio* (that indicates if the patient has any cardiovascular disease) and *Onco*. It is interesting to remark that this logistic regressor has identified the variable *Onco* as important. The decision tree algorithm could identify it as important as well in another branch, but not in this one, showing the improvements that MLTree brings by adding other classification models to tree leaves.

The age and comorbidities values chosen by MLTree are conditions recognized as risk factors in COVID-19 patients by institutions like the Centers for Disease Control and Prevention [20][21]. This exhibits that MLTree has extracted from the data

```
1 Root
|   2 Age < 54.5
|   |   3 Onco < 0.5: LogisticRegression(solver='liblinear')
|   |       |              [[ 0.09622804, 0., 1.06310088, 0.02720569, -0.48114559, 0.56860273, -0.25089506, 0.09046084, 0., -0.86769405, 0.13086652, -0.0719601, -0.0905102 ]]
|   |   4 Onco >= 0.5: True
|   5 Age >= 54.5: LogisticRegression(solver='liblinear')
                      [[ 0.04523051, 0., -0.05810508, 0.27041788, 0.13007256, 0.17136809, 0.15520303, 0.03358627, 0.32898076, -0.0965259, -0.00612677, -0.02895031, -0.0266998 ]]
```

Fig. 2 MLTree model using Logistic Regression and the dataset emergency.

information that matches reality and it demonstrates that the model could be used by doctors in their hospitals.

V. Conclusion

The main objective of this paper was to contribute to the fight against the pandemic caused by the COVID-19 disease. We presented MLTree, a new machine learning algorithm that generates a decision tree extended with *other classification algorithms* in some of its leaves to improve the accuracies obtained using each algorithm by itself. We showed that MLTree is able to generate good solutions that can help doctors in their process of deciding which patients that come to the emergency department need an admission and which ones can stay at home. It was also demonstrated that MLTree generates models that are significantly different from the ones a common decision tree would create, and the accuracies calculated show that it is possible that they are outperforming them. Moreover, in the majority of the experiments performed, MLTree generates models that are at least equally and sometimes even better that the ones each base classification algorithm would generate by itself. Finally, we proposed an MLTree model with one of the configurations used in the experiments as a viable solution, and an interpretation of it was given in which we assessed that it matched reality and thus it could be used in a real scenario. Moreover, this interpretation demonstrates that, as long as MLTree uses other interpretable classification algorithms, it keeps the high understandability that is brought by decision trees.

This new tool to help in the fight against COVID-19 considers a hypothetical situation in which the cost of making an error in the prediction is the same for all class labels. However, the cost of sending an ill patient home might be higher than admitting a healthy one. A proper continuation of this work could be the discussion of a cost function to assign weights to each label: critical and non-critical, in order to make the model to classify with less error one of them.

As MLTree is based on the Classification and Regression algorithm (CART) [15], possible future research could also be aimed at adding the functionality to make it able to solve regression problems.

Finally, even though hospital overcrowding isn't a major issue anymore, we highly believe that the algorithm and the approach presented in this paper could be used as an inspiration to solve similar situations in the future that may occur with other diseases.

Acknowledgments

This work has been partially supported by the project funded by Fundación BBVA: "Outcome prediction and treatment efficiency in patients hospitalized with COVID-19 in Madrid: A Bayesian network approach", in which the dataset preprocessing was performed, by Fundación Jiménez Díaz, that provided us with the necessary data for this study, and by "Red Temática sobre Inteligencia Artificial en Biomedicina (RED2018-102312-T)" of the Ministry of Science and Innovation.

References

[1] World Health Organization, "WHO Director-General's remarks at the media briefing on 2019-nCoV on 11 February 2020". Available: http://ow.ly/Yvxs50FfHAz.

[2] World Health Organization, "WHO Coronavirus (COVID-19) dashboard" . Available: https://covid19.who.int/.

[3] G.A. Roth, S. Emmons-Bell, H.M. Alger, S. M. Bradley, S.R. Das, J.A. de Lemos, et al., "Trends in patient characteristics and COVID-19 in-hospital mortality in the United States during the COVID-19 pandemic", 2021

[4] R.C. Barros, A.T Winck, K.S. Machado, M.P. Basgalupp, A. CPLF de Carvalho, D.D. Ruiz, O.N. de Souza, "Automatic design of decision-tree induction algorithms tailored to flexible-receptor docking data", 2012

[5] J. R. Quinlan, «C4.5: Programs for machine learning».

[6] L. Wei-Yin, "Classification and regression trees", *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 2011.

[7] J.L. Izquierdo, J. Ancochea, Savana COVID-19 Research Group, J.B. Soriano, "Clinical characteristics and prognostic factors for intensive care unit admission of patients with COVID-19: retrospective study using machine learning and natural language processing" 2020.

[8] M. Pourhomayoun, M. Shakibi, "Predicting mortality risk in patients with COVID-19 using machine learning to help medical decision-making", 2021.

[9] R. Kohavi, "Scaling up the accuracy of naive-Bayes classifiers: a decision-tree hybrid", 2011.

[10] J.Gama, P. Brazdil, "Cascade generalization", 2000.

[11] Z. Zhi-Hua, C. Zhao-Qian, "Hybrid decision tree", 2002.

[12] N. Landwehr et al., "Logistic model trees", 2005.

[13] S. Gil-Begue, P. Larrañaga, C. Bielza ,"Multi-dimensional Bayesian network classifier trees", 2018.

[14] L. Todorovski, S. Džeroski, "Combining classifiers with meta decision trees", 2003.

[15] L. Breiman, J. Friedman, C.J. Stone, R.A. Olshen, "Classification and regression trees", 1984.

[16] J. Valente, "Decision tree from scratch in Python". Available: http://ow.ly/gNo550FchG5.

[17] J. Hoare, "Machine learning: pruning decision trees".

[18] Pedregosa et al., "Scikit-learn: Machine Learning in Python", JMLR 12, pp. 2825-2830, 2011.

[19] T.G. Dietterich, "Approximate Statistical Tests for Comparing supervised classification learning algorithms".

[20] Centers for Disease Control and Prevention, "People with certain medical conditions". Available: http://ow.ly/a3kC50EXvwv

[21] Centers for Disease Control and Prevention, "Weekly updates by select demographic and geographic characteristics". Available: https://www.cdc.gov/nchs/nvss/vsrr/covid_weekly/index.html

Cost-effectiveness analysis with decision analysis networks

Francisco Javier Díez[1,2] Manuel Luque[1,2] Manuel Arias[1,2] Jorge Pérez-Martín[1,2]
[1]*Dept. Artificial Intelligence*
Universidad Nacional de Educación a Distancia (UNED)
Madrid, Spain
{fjdiez,mluque,marias,jperezmartin@dia.uned.es}
[2]*DeciSupport AI*
Madrid, Spain

***Abstract*—Decision analysis networks (DANs) are a new type of probabilistic graphical model. Like influence diagrams (IDs), they are much more compact and easier to build than decision trees and can represent conditional independencies, but they can also represent problems involving partial orderings of the decisions (order asymmetry) and other types of asymmetry. Given that DANs can solve symmetric problems as easily and as efficiently as IDs, and are more appropriate for asymmetric problems—which include virtually all real-world problems—DANs might replace IDs as the standard type of probabilistic graphical model for decision analysis. This paper describes the application of DANs to cost-effectiveness analysis (CEA), which is used increasingly in medicine to determine whether the health benefit of an intervention is worth the economic cost.**

***Index Terms*—Decision analysis networks, decision trees, influence diagrams, probabilistic graphical models, medical decision making, cost-effectiveness analysis.**

I. Introduction

A. Probabilistic models for decision analysis

A probabilistic graphical model (PGM) [1] consists of a probability distribution P, defined on a set of variables $\mathbf{X}$, and a graph, such that each node in the graph represents a variable and—roughly speaking—edges represent dependencies among those variables; more precisely, every absent link denotes a relation of probabilistic independence. PGMs include include Bayesian networks [2], influence diagrams (IDs) [3], and several types of Markov models, and have been extensively studied in the field of artificial intelligence.

The two formalisms most widely used for the representation and analysis of decision problems are decision trees (DTs) [4] and IDs. DTs have the advantage of almost absolute flexibility, but also have several drawbacks, mainly that their size grows exponentially: an ID whose chance and decision nodes represent n binary variables is equivalent to a decision tree with around 2^n leaves. In contrast, IDs have the advantages of being very compact, easily representing conditional independence, and using direct probabilities, which allows them to address much larger problems. For example, Luque et al. [5] built Mediastinet, an ID for finding the optimal sequence of tests for the mediastinal staging of non-small cell lung cancer. Building an equivalent DT would have been impossible because the tree would have more than 10,000 branches.

The main limitation of IDs is that they can only represent symmetric decision problems [6]. There is *domain asymmetry* when the value taken by a variable X restricts the values that variable Y can take (for example, the result of a test Y is positive or negative only when the decision X is to do the test); there is *information asymmetry* when a variable Y (for example, the result of a test) is observed for some values of a variable X (in our example, when X = 'do the test') but not for others ('do not test'); and there is *order asymmetry* when the decisions can be made in different orders (for example, when the order of several tests is not set a priori). Virtually all real-world problems are asymmetric. For example, when addressing the mediastinal staging of lung cancer the pneumologist was uncertain about the optimal ordering of the tests.

For this reason, Díez et al. [6] introduced *decision analysis networks* (DANs), which, in addition to representing all symmetric problems as easily as with IDs, can also represent and solve asymmetric problems. In particular, the DAN version of Mediastinet did not need a total ordering of the tests.

B. The difficulty of cost-effectiveness analysis

Due to the rapid increase of medical expenditures in all countries, every health system must analyse whether the benefit of each intervention outweighs its economic cost. Cost-effectiveness analysis (CEA), the most common type of economic evaluation, is based on the notion of *net monetary benefit* [7],

$$NMB(\lambda) = \lambda \cdot e - c\,, \tag{1}$$

where e is the average effectiveness and c the average cost of the intervention for the subpopulation of interest. The parameter λ, usually called *willingness to pay* or *cost-effectiveness threshold*, converts effectiveness into a

Supported by the Spanish Ministry of Science and Innovation under grant PID2019-110686RB-I00.

monetary scale. It takes values on the set of positive real numbers, i.e., on the interval $(0, +\infty)$, and is measured in effectiveness units divided by cost units; for example, in dollars per death avoided or euros per quality-adjusted life year (QALY) [8].

When there are several incompatible interventions for a problem, CEA looks for the one having the highest NMB. If no intervention dominates the others (i.e., no one is more effective and cheaper than the others), the optimal intervention depends on λ.

As the willingness to pay is specific for each decision maker—in this case, for each health system—and often imprecise or uncertain, CEA must consider all the possible values of λ. It may conclude, for example, that when the willingness to pay is below 35,847 euros per QALY the optimal intervention would be not to do anything, and when it is above that threshold the optimal intervention would be to do a test and, if positive, apply a certain therapy.

The main method that health economists use for CEA are DTs [9], in spite of their serious limitations. In particular, the size of the tree, which grows exponentially in the unicriterion case, as mentioned above, grows much faster for CEA, because the standard algorithm [4] cannot evaluate decision trees with embedded decision nodes, which are those other than the root of the tree [10], [11].

Despite of the increasing importance of economic evaluations for medical decision making, artificial intelligence has not addressed this problem until very recently, due to the lack of appropriate tools. In particular, the standard algorithms for IDs [12], [13], [14] could only evaluate unicriterion models, so they were unable to perform CEA. For example, they could evaluate the ID Mediastinet for each single value of λ, but they could not find a set of n thresholds delimiting $n + 1$ intervals for the willingness to pay, with an optimal intervention for each interval. In contrast, the algorithm by Arias et al. [15] was able to perform CEA with IDs, thus being the first application of an AI method to the economic evaluation of health technologies.

However, that algorithm still had one drawback, stemming from the inability of IDs to represent asymmetric problems, so every ID requires a total ordering of the decisions. In the case of mediastinal staging of lung cancer, the problem was that neither the pulmonologist could determine the optimal order with certainty, nor the ID was able to find it.

II. Cost-effectiveness analysis with DANs

The problem was solved by designing a CEA algorithm for DANs [16] that combined the method for evaluating unicriterion DANs [6] with the CEA algorithm for decision trees with embedded decision nodes [17]. The former generates a tree of symmetric DANs and the latter evaluates it from the leaves to the root, generating a set of thresholds and intervals for the willingness to pay, each interval having a value of effectiveness, a cost, and an optimal intervention, just as in the evaluation of IDs [15]. The basic idea was straightforward, but integrating those techniques required significant effort.

DANs and the CEA algorithms have been implemented in OpenMarkov, an open-source tool for building and evaluating probabilistic graphical models, which has been used for teaching and research in more than 30 countries, from top universities, research centres of the US Government and large companies, to teachers and students in poor countries, who could not afford the price of commercial products for probabilistic graphical models or for cost-effectiveness analysis.

References

[1] D. Koller and N. Friedman, *Probabilistic Graphical Models: Principles and Techniques*. Cambridge, MA: The MIT Press, 2009.

[2] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Mateo, CA: Morgan Kaufmann, 1988.

[3] R. A. Howard and J. E. Matheson, "Influence diagrams," in *Readings on the Principles and Applications of Decision Analysis* (R. A. Howard and J. E. Matheson, eds.), p. 719–762, Menlo Park, CA: Strategic Decisions Group, 1984.

[4] H. Raiffa and R. Schlaifer, *Applied Statistical Decision Theory*. Cambridge, MA: John Wiley & Sons, 1961.

[5] M. Luque, F. J. Díez, and C. Disdier, "Optimal sequence of tests for the mediastinal staging of non-small cell lung cancer," *BMC Medical Informatics and Decision Making*, vol. 16, p. 1–14, 2016.

[6] F. J. Díez, M. Luque, and I. Bermejo, "Decision analysis networks," *International Journal of Approximate Reasoning*, vol. 96, p. 1–17, 2018.

[7] A. A. Stinnett and J. Mullahy, "Net health benefit: A new framework for the analysis of uncertainty in cost-effectiveness analysis," *Medical Decision Making*, vol. 18, p. S68–80, 1998.

[8] M. C. Weinstein, G. Torrance, and A. McGuire, "QALYs: The basics," *Value in Health*, vol. 12, p. S5–S9, 2009.

[9] M. F. Drummond, M. J. Sculpher, K. Claxton, G. L. Stoddart, and G. W. Torrance, *Methods for the Economic Evaluation of Health Care Programmes*. Oxford, UK: Oxford University Press, 4th ed., 2015.

[10] K. M. Kuntz and M. C. Weinstein, "Modelling in economic evaluation," in *Economic Evaluation in Health Care* (M. F. Drummond and A. McGuire, eds.), ch. 7, p. 141–171, New York: Oxford University Press, 2001.

[11] M. Arias and F. J. Díez, "The problem of embedded decision nodes in cost-effectiveness decision trees," *Pharmacoeconomics*, vol. 32, p. 1141–1145, 2014.

[12] S. M. Olmsted, *On Representing and Solving Decision Problems*. PhD thesis, Dept. Engineering-Economic Systems, Stanford University, CA, 1983.

[13] R. D. Shachter, "Evaluating influence diagrams," *Operations Research*, vol. 34, p. 871–882, 1986.

[14] M. Luque and F. J. Díez, "Variable elimination for influence diagrams with super-value nodes," *International Journal of Approximate Reasoning*, vol. 51, p. 615–631, 2010.

[15] M. Arias and F. J. Díez, "Cost-effectiveness analysis with influence diagrams," *Methods of Information in Medicine*, vol. 54, p. 353–358, 2015.

[16] F. J. Díez, M. Luque, M. Arias, and J. Pérez-Martín, "Cost-effectiveness analysis with unordered decisions," *Artificial Intelligence in Medicine*, vol. 117, p. 102064, 2021.

[17] M. Arias and F. J. Díez, "Cost-effectiveness analysis with sequential decisions," Technical Report CISIAD-11-01, UNED, Madrid, Spain, 2011.

XIX Conferencia de la Asociación Española para la Inteligencia Artificial

I Taller de IA aplicada a los Sistemas Inteligentes de Transporte (IA-SIT)



Cameras vs LiDAR using Deep Learning

1st Fernando Rojas Ramos
Facultad de Ciencias de la Computación
Benemérita Universidad Autónoma de Puebla
Puebla, México
toshk15@hotmail.com

2nd Ivo H. Pineda Torres
Facultad de Ciencias de la Computación
Benemérita Universidad Autónoma de Puebla
Puebla, México
ivopinedatorres@gmail.com

***Abstract*—Research centers and companies dedicated to the development of autonomous vehicles are opting for two trends: using only cameras as a vision system or using LiDAR sensors plus cameras. Tesla has a fully camera vision system and Waymo has a cameras and LiDAR system. The difference of the LiDAR sensor could prevent accidents and save human lives in the future. Therefore, the main contribution of this work is the design of a methodology based on the comparison of detection efficiency for vision devices (Cameras and LiDAR). Applying measurement parameters provided by Neural Networks and models evaluation metrics in machine learning; it has to be concluded if its necessary to use LiDAR sensors in the development of autonomous cars.**

***Index Terms*—Autonomous Vehicles, Vision System, Neural Networks.**

I. Introduction

Currently, the development of autonomous cars is of great interest to most automotive companies, including Tesla, Google, VW, Toyota, Ford and many others. The Waymo company provides a robot taxi service in the city of Phoenix in the United States; this is a limited service for this region and with their cars being equipped with a vision system based on cameras, radar and LiDAR sensors. Their aim is to autonomously keep their vehicles centered on the lane and to reach the passenger's destination.

Moreover, the Tesla company is also in the fight to achieve level 5 autonomy for its cars, which include advanced hardware that currently provides Autopilot functions, and full autonomous driving functionalities in the near future through updates. While their software is designed to improve functionality as time goes on, it has, however run, into many difficulties with a considerable number of car accidents. Put on doubt its vision system based only on cameras and radar-type sensors, Elon Musk CEO of Tesla is against using a LiDAR sensor.

Tesla has been heavily relying on vision and going against LiDAR sensors. Simultaneously, all the other companies use LiDAR and seem to dismiss other options.

The most apparent reason for Tesla has taken a different route is the cost. The cost of placing a single LiDAR device on a car is somewhere around $10,000. Google with its Waymo project has been able to slightly decrease the number by introducing mass production. However, the cost is still rather significant.

Car accidents are the eighth leading cause of death worldwide with 95% being caused by human error; the expectation is that the automation of transport represents a significant reduction in the number of occurrences and mainly of victims.

Nowadays, many current image-based object detectors using convolutional neural networks exhibit excellent performance on existing datasets such as KITTI [1]. This dataset will be used to carry out the study with images and real data from LiDAR sensors.

LiDAR is an acronym for light detection and ranging, has a variety of applications, including soil science, meteorology, seismology, military applications, geology and atmospheric physics among others. A typical LiDAR sensor emits pulsed light waves into the surrounding environment. These pulses bounce off surrounding objects and return to the sensor. The sensor uses the time it took for each pulse to return to the sensor to calculate the distance it traveled. Repeating this process millions of times per second creates a precise, real-time 3D map of the environment. LiDAR can detect the distance to an object and apply the brakes to slow down the car. LiDAR has been a proven technology in measuring distance and it is for this reason engineers have used it for different applications including self-driving cars.

Instead of using light pulses like LiDAR, the cameras use visual data returned from the optics in the lens to an on-board software for further analysis. Using neural networks and computer vision algorithms, objects can be identified to provide the car information while it is driving. This helps the car avoid collisions, slow down when there is traffic, safely make lane changes and even read the text from signs on the road or highway using Optical Character Recognition. So far Tesla has shown that self-driving cars can perform without LiDAR by using cameras.

The main contributions of this work is the development of a result evaluation-based methodology to compare the detection efficiency of both devices (Cameras and LiDAR). who are based on the measurement parameters provided by neural networks and model evaluation metrics in machine learning.

This paper is organized as follows: Section 2 gives a brief analysis of the most related papers. Section 3 proposes methodology to analyze the performance of the built detectors. Finally, the conclusions and future work in Section 4.

II. Related Work

There are a variety of studies aimed at the autonomy of automobiles and, in particular, at devices that function as a

means of vision-in this case LiDAR sensors and Cameras. The studies related to this project are focused on examining the detection of objects separately, that is, the results are based on the detection of objects using only cameras [2]–[5] or LiDAR sensors [6]–[9]. In this project, the two trends are covered to carry out a methodology for comparing the results. This serves to determine which device provides the best detection.

Manuel Herzog and Klaus Dietmayer [10], propose in their work the detection of objects with LiDAR sensors in driverless cars and by applying the strategy of training a model using data from different types of LiDAR sensors. In comparison to the author Haris, M. [11]: this work proposes the detection of small and medium obstacles that were left on the road intentionally or unintentionally, which can pose a danger for both autonomous and human driving situations. They use Random Markov Field (MRF) models by merging three potentials (gradient potential, curvature prior potential, and depth variance potential) to segment obstacles and non-obstacles in the hazardous environment. Finally, they use obstacle detection to predict the steering wheel angle of the autonomous car using images from cameras.

Di Feng [12]: he tries to summarize systematically the methodologies in his article and to discuss the challenges for deep multimodal object detection and semantic segmentation in autonomous driving. To this end, they first provided an overview of sensors embedded in test vehicles, open datasets, and basic information for object detection and semantic segmentation in autonomous driving research.

The technological trend in using sensor fusion for the development of autonomous cars is of great interest and study for the coming years because it involves the integration of all data from radars, LiDAR, cameras, gps, etc.

Guan [13]: propose in his work the fusion of sensors to achieve better performance in the detection of objects. This consists of a multi-adaptive and high precision completion method which improves the adaptability to the detection environment and performs preliminary fusion of data from two sensors (Camera and LiDAR). The system realized fast and accurate object detection through the real-time object detection model called YOLOv3 and applied a proposed decision-level fusion strategy. The methodology not only gets higher detection precision during daytime driving but also obtains the distance between the front vehicle and the detecting vehicle.

Wangs' work [14] is based on the use of stereo cameras for the detection of objects, these images have the advantage of showing more information than the images of monocular cameras; they use depth in image scenes to locate how far away the detected object is as with LiDAR sensors. With these images they are able to simulate the operation of the LiDAR sensors. However, they also prove that the fusion of the images from the cameras plus the point cloud of the LiDARs obtain better detection results. The article proposes as future work to compare the processing times of the data from both detection devices.

III. Proposed Methodology

In order to solve the problem of object detection using Deep Learning, the methodology shown in Fig. 1 is proposed. A convolutional neural network called U-net is used for the segmentation of labeled objects in the dataset KITTI for their subsequent detection in a bounding box and is also used the convolutional neural network called YOLOv5 recently published to compare the detection results, which is the objective of this work.

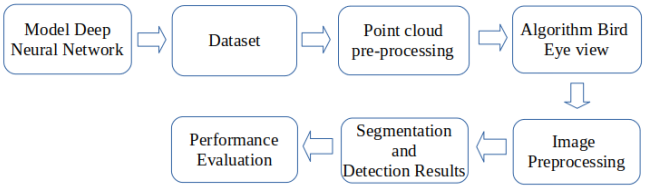


Fig. 1. Proposed Methodology

A. *Models Deep Neural Networks U-net and YOLOv5*

The U-Net was developed by Olaf Ronneberger [15] for the segmentation of biomedical images. The architecture contains two paths. The first path is the shrink path (also called the encoder) that is used to capture the context in the image. The encoder is just a traditional stack of maximum grouping and convolutional layers. The second path is the symmetric expansion path (also called a decoder) that is used to allow precise localization by transposed convolutions. So it is a fully convolutional end-to-end network (FCN), that is, it only contains convolutional layers and does not contain any dense layers due to which it can accept images of any size.

The YOLOv5 [16] got released by Glenn Jocher (Founder and CEO of Utralytics) and has been chosen for the task of object detection and for its incredible characteristics: the speed and excellent precision in the detection of objects. These are the following versions of YOLO with enhancements for the detection of small objects: YOLOv1, YOLOv2, YOLOv3, YOLOv4 and the latest recent YOLOv5 release. This model has the capacity to process 140 frames per second with the disadvantage of the first version of not accurately detecting small objects, this problem has been improved with the development of the existing versions. YOLO has an average precision (mAP) value of 57.9% on the COCO dataset, which is significantly higher than an SSD type network and a RetinaNet, being 4 times faster than them, 100 times faster than a Fast R-CNN.

B. *KITTI Dataset*

KITTI [1] is a dataset available to carry out the study with real data of a prototype autonomous vehicle. The vehicle is equipped with four cameras: 1 stereo pair of color cameras and 1 stereo pair of grayscale cameras. The color and grayscale cameras are mounted close to each other (6 cm) the baseline of both stereo decks is approximately 54 cm. This configuration allows to obtain information in both color and grayscale from

the left and right camera. While color cameras (obviously) come with color information, grayscale camera images have higher contrast and slightly less noise.

All cameras are clocked at approximately 10 Hz relative to the Velodyne laser scanner. The trigger is mounted in such a way that the images from the camera roughly coincide with Velodyne lasers facing forward (in the driving direction).

All camera images are provided as lossless compressed and rectified png sequences. Native image resolution is 1382x512 pixels and slightly lower after rectification.

The classes available in the KITTI dataset are 8 and the number of instances are the following, Car = 28614, cyclist = 612, Misc = 959, Pedestrian = 4448, Person sitting = 220, Tram = 504, Truck = 1079, Van = 2900.

C. Point cloud preprocessing

Point clouds are a collection of points that represent a 3D shape or feature. Each point has its own set of X, Y and Z coordinates and in some cases additional attributes.

Nowadays, object detection systems can be divided into two main categories . The first ones are the geometric based, which retrieve the obstacles using geometric and morphological operations on the 3D points. The seconds are the deep learning based, which process the 3D points, or an elaboration of the 3D point cloud, with deep learning techniques to retrieve a set of obstacles.

This work is focused on the second approximation: projection based methods implement a single or multi-view projection of a 3D point cloud, resulting in a 2D grid, which is then processed to find object clusters with the desired confidence. Afterwards, this grid is processed by is then processed by a 2D Convolution neural network.

The sensors setup used for the rigid transformation between camera and LiDAR. The dimensions and mounting positions of the sensors with respect to the vehicle body. The height of the Velodyne laser scanner is 1.73 m and the height of the cameras is 1.65 m measured with respect to the road surface. The Velodyne HDL-64E rotating 3D laser scanner, 10 Hz, 64 beams, 0.09 degree angular resolution, 2 cm distance accuracy, collecting 1.3 million points/second, field of view: 360° horizontal, 26.8° vertical, range: 120 m. 2 PointGray Flea2 color cameras (FL2-14S3C-C), 1.4Megapixels, 1/2 Sony ICX267 CCD, global shutter.

The first thing to do is bring the 3D point cloud of the LiDAR sensors to a 2D voxel type image, with the alignment of the points in space and time according to the calibration algorithm of the cameras with the LiDAR sensor. The calibration algorithm calculates the camera matrix using the extrinsic and intrinsic parameters. The extrinsic parameters represent a rigid transformation from 3D world coordinate system to the 3D cameras coordinate system. The intrinsic parameters represent a projective transformation from the 3D cameras coordinates into the 2D image coordinates. As shown in our diagram see Fig. 2, we have the image of the camera and the point cloud of the LiDAR sensors; the objective is the synchronization of the points with the pixels of the image so that the 2D surface of our new image coincides with the projection of the 3D point cloud. Using the camera's calibration parameters, we can carry out the transformation of the point cloud to a 2D voxel image as the camera does.

D. Algorithm Bird eye view

The perspective transform that interests us is a birds-eye view transform [17] that enables us to view a lane from above. Aside from creating a birds eye view representation of an image, a perspective transform can also be used for all kinds of different view points.

Channel feature extraction [18] the 3D point-cloud provided by the LiDAR is projected in a BEV image with predetermined width, height and grid cell size. To avoid loss of information during the projection of 3D point-cloud into a 2D image, 6 additional channels are stacked together the new pattern to recover information about the peak and the medium values of height, intensity and distribution of the collapsed points for each cell. Moreover, binary information concerning the effective occupancy of each grid are included.

E. Image Preprocessing

This phase involves the preprocessing of digital images to provide them with different attributes with data augmentation [19], [20]. In the real world scenario, it's possible that the existing datasets are taken under a limited conditions. This is why data augmentation is applied to simulate different conditions and generate a random variety images.

Popular augmentation techniques: lightening the image to increase their clarity and avoiding dark images; decreasing the contrast to reduce the total color range of the image; applying a Gaussian filter to eliminate noise and details of the texture; flipping images horizontally and vertically; rotating it at right angles will preserve the image size; scaled outward or inward; the method of resizing the section is popularly known as random cropping; translation just involves moving the image along the X or Y direction (or both).

F. Segmentation and Detection Results

According to the design of the methodology, the necessary phases have been implemented to obtain the results of segmentation and detection in images of the cameras Fig. 3 and images from the data of the LiDAR sensors Fig. 4. In this phase the experiments are carried out successfully with the Car class which has the highest number of labels and, therefore, presents balance with respect to the rest of the classes available in the KITTI dataset.

The processes described in the methodology have been run on a computer with a GeFORCE GTX 1050 graphics card and 24 GB of RAM necessary for the process of generating the images with bird eye view from the point cloud of the LiDAR sensors; this takes extra time compared to the images from cameras. This phase of the methodology is of primary interest-working with other types of sensor data to train a neural network. The results present four images per row as follows in this case for the neural network U-net: The first

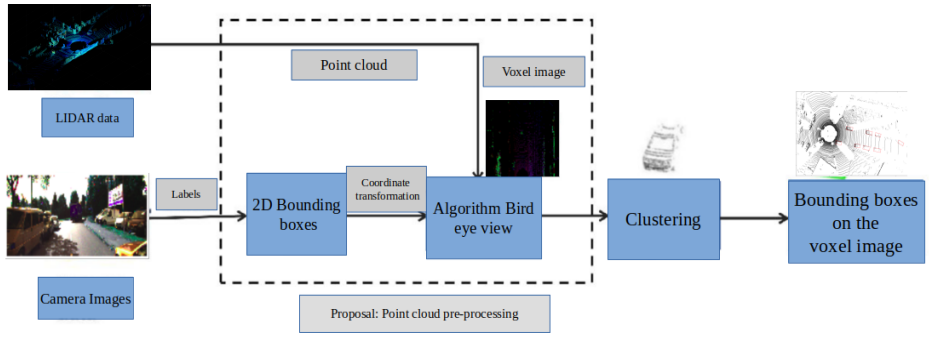


Fig. 2. The coordinate conversion between the image and LiDAR



Fig. 3. Segmentation and detection of the ”Car” class with Camera.

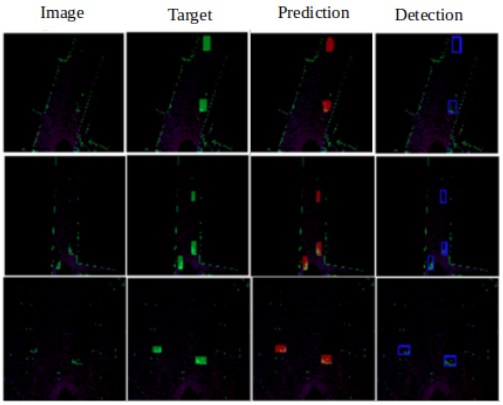


Fig. 4. Segmentation and detection of the ”Car” class with LiDAR.

image represents the original test image; the second image shows the total targets to be predicted; the third image shows the prediction of the object's segmentation by the U-net neural network; ultimately, the fourth image with a bounding box represents the detection of the segmentation predicted by the neural network with a threshold greater than or equal to 0.5.

The following results are those obtained with the convolutional neural network YOLOv5. The object detection is performed directly on the test images with a bounding box and a label with the name of the class. During training, the YOLOv5 training pipeline creates batches of training data with augmentations. We can visualize the training data ground truth as well as the augmented training data. The first figure 5 shows the results obtained with the images of the cameras and the second figure 6 shows the results with the LiDAR sensor.



Fig. 5. Detection of the ”Car” class with Camera.

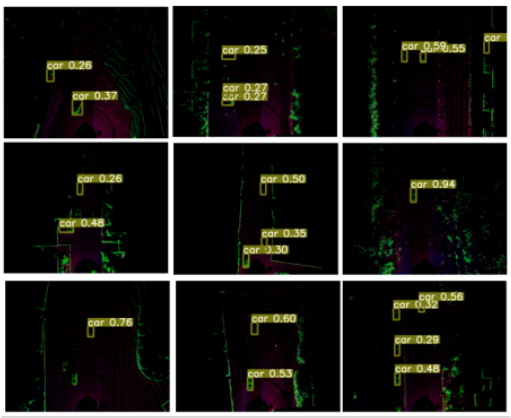


Fig. 6. Detection of the ”Car” class with LiDAR.

G. *Performance Evaluation*

Here are three metrics used as reference; these are precision, recall, and measure F1, which assess the balance between precision and recall.

$$precision = \frac{TruePositives}{TruePositives + FalsePositives} \quad (1)$$

$$recall = \frac{TruePositives}{TruePositives + FalseNegatives} \quad (2)$$

$$F1\,measure = 2 \times \frac{precision \times recall}{precision + recall} \quad (3)$$

Below are the confusion matrix in the tables I and II showing the evaluation corresponding to the predictions of the two deep neural networks with different test sets.

TABLE I
TEST SET 100 IMAGES AND TOTAL 261 INSTANCES.

Model	TP	FN	TN	FP
YOLOv5 Camera	176	85	0	0
YOLOv5 LiDAR	144	48	0	0
U-net Camera	123	45	53	40
U-net LiDAR	132	42	55	38

TABLE II
TEST SET 200 IMAGES AND TOTAL 482 INSTANCES.

Model	TP	FN	TN	FP
YOLOv5 Camera	336	142	0	4
YOLOv5 LiDAR	264	84	0	2
U-net Camera	233	88	99	82
U-net LiDAR	254	83	98	80

The following tables III and IV, shows the evaluation metrics with respect to the predictions made by deep neural networks U-net and YOLOv5. It is important to mention that at this point the only class that is being evaluated is ”Car”, as described it is the class with the largest number in instances, unlike the other classes available in the dataset.

TABLE III
TEST SET 100 IMAGES AND TOTAL 261 INSTANCES.

Model	Presicion	Recall	F1 measure
YOLOv5 Camera	100.00%	67.00%	80.00%
YOLOv5 LiDAR	100.00%	75%	85.71%
U-net Camera	75.46%	73.21%	74.31%
U-net LiDAR	77.64%	75.80%	76.70%

TABLE IV
TEST SET 200 IMAGES AND TOTAL 482 INSTANCES.

Model	Presicion	Recall	F1 measure
YOLOv5 Camera	98.8%	70.00%	81.90%
YOLOv5 LiDAR	99.2%	75.80%	85.90%
U-net Camera	73.96%	72.58%	73.26%
U-net LiDAR	76.04%	75.37 %	75.70%

IV. Conclusion and Future Work

In this work, it was possible to integrate advanced computer vision algorithms using deep neural networks, whose results were positive in recent studies conducted in autonomous prototype vehicles.

Data from the LiDAR sensors were analyzed which provides us with a 360 degree view with a single sensor compared to cameras. The LiDAR sensor provides good results in the segmentation and detection of objects; therefore, the proposal is to use it in conjunction with the cameras to develop a sophisticated and complete vision system. This means, that if at some point the cameras do not detect an object while driving, the LiDAR sensor is supported as security.

The depth estimation and the creation of maps of the LiDAR sensors help to detect and locate the objects within the environment. These characteristic could allow to avoid accidents such as those that have occurred with the Tesla brand cars.

In this work, we conclude the efficiency of the LiDAR sensor for its instrumentation in autonomous cars.

As future work, it is proposed to analyze the possibility of equipping a car with a LiDAR sensor and cameras to generate its own dataset on the roads of Mexico.

Acknowledgment

We gratefully acknowledge to Conacyt with the financial support to carry out this research.

References

[1] Geiger, A., Lenz, P., and Urtasun, R. Are we ready for autonomous driving? the kitti vision benchmark suite. In 2012 IEEE Conference on Computer Vision and Pattern Recognition (pp. 3354-3361). IEEE.

[2] Ramesh, B., Ussa, A., Della Vedova, L., Yang, H., and Orchard, G. Low-power dynamic object detection and classification with freely moving event cameras. Frontiers in neuroscience, 14, 135, 2020.

[3] Meyer, M., and Kuschk, G. Deep learning based 3d object detection for automotive radar and camera. In 2019 16th European Radar Conference (EuRAD) (pp. 133-136). IEEE.

[4] Nobis, F., Geisslinger, M., Weber, M., Betz, J., and Lienkamp, M. A deep learning-based radar and camera sensor fusion architecture for object detection. In 2019 Sensor Data Fusion: Trends, Solutions, Applications (SDF) (pp. 1-7). IEEE.

[5] Fuentes-Jimenez, D., Martin-Lopez, R., Losada-Gutierrez, C., Casillas-Perez, D., Macias-Guarasa, J., Luna, C. A., and Pizarro, D. DPDnet: A robust people detector using deep learning with an overhead depth camera. Expert Systems with Applications, 146, 113168, 2020.

[6] Chen, C., Fragonara, L. Z., and Tsourdos, A. RoIFusion: 3D Object Detection From LiDAR and Vision. IEEE Access, 9, 51710-51721, 2020.

[7] Meyer, G. P., Laddha, A., Kee, E., Vallespi-Gonzalez, C., and Wellington, C. K. Lasernet: An efficient probabilistic 3d object detector for autonomous driving. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 12677-12686), 2019.

[8] Lang, A. H., Vora, S., Caesar, H., Zhou, L., Yang, J., and Beijbom, O. Pointpillars: Fast encoders for object detection from point clouds. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 12697-12705), 2019.

[9] Sahba, R., Sahba, A., Jamshidi, M., and Rad, P. 3D Object Detection Based on LiDAR Data. In 2019 IEEE 10th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON) (pp. 0511-0514). IEEE.

[10] Herzog, M., Dietmayer, K. Training a fast object detector for lidar range images using labeled data from sensors with higher resolution. In 2019 IEEE Intelligent Transportation Systems Conference (ITSC) (pp. 2707-2713). IEEE.

[11] Haris, M., and Hou, J. (2020). Obstacle detection and safely navigate the autonomous vehicle from unexpected obstacles on the driving lane. Sensors, 20(17), 4719.
[12] Feng, D., Haase-Schuetz, C., Rosenbaum, L., Hertlein, H. and Dietmayer, K. Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges. IEEE Transactions on Intelligent Transportation Systems, 2020.
[13] Guan, L., Chen, Y., Wang, G., and Lei, X. Real-Time Vehicle Detection Framework Based on the Fusion of LiDAR and Camera. Electronics, 9(3), 451, 2020.
[14] Wang, Y., Chao, W. L., and Garg, D. Pseudo-lidar from visual depth estimation: Bridging the gap in 3d object detection for autonomous driving. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 8445-8453), 2019.
[15] Ronneberger, O., Fischer, P., and Brox, T. U-net: Convolutional networks for biomedical image segmentation. In International Conference on Medical image computing and computer-assisted intervention (pp. 234-241). Springer, Cham, 2015.
[16] https://github.com/ultralytics/yolov5
[17] Yang, G., Mentasti, S., Bersani, M., Wang, Y., Braghin, F., and Cheli, F. LiDAR point-cloud processing based on projection methods: a comparison. In 2020 AEIT International Conference of Electrical and Electronic Technologies for Automotive (AEIT AUTOMOTIVE) (pp. 1-6). IEEE.
[18] Lee, K. H., Kliemann, M., Gaidon, A., Li, J., Fang, C., Pillai, S., and Burgard, W. PillarFlow: End-to-end Birds-eye-view Flow Estimation for Autonomous Driving. arXiv e-prints 2020, arXiv-2008.
[19] Tian, Y., Pei, K., Jana, S., and Ray, B. Deeptest: Automated testing of deep-neural-network-driven autonomous cars. In Proceedings of the 40th international conference on software engineering (pp. 303-314), 2018.
[20] Du, S., Guo, H., and Simpson, A. Self-driving car steering angle prediction based on image recognition, 2019. arXiv preprint arXiv:1912.05440.

Optimización del entorno de simulación para el depurado de arquitectura software en vehículos autónomos

David Yagüe-Cuevas*, Alejandro Ruíz-de-la-Cuadra†, Pablo Marín-Plaza‡ and José María-Armingol§
Laboratorio de Sistemas Inteligentes (LSI), Universidad Carlos III de Madrid (UC3M)
Leganés, Madrid, España
Email: *dayaguec@inf.uc3m.es, †aruizla@inf.uc3m.es, ‡pamarinp@ing.uc3m.es, §armingol@ing.uc3m.es

***Abstract*—La movilidad autónoma es una de las áreas tecnológicas actuales que ha sufrido gran cantidad de avances en poco tiempo. El interés general del mundo académico así como el industrial ha sabido encontrar un nicho suficientemente interesante y complejo para su estudio, a la par que potencialmente beneficioso para la industria y la sociedad. Una de sus particularidades es la necesidad de generar sistemas seguros y fiables, que exigen poseer un entorno válido para realizar aquellas pruebas que avalen el cumplimiento de los requisitos propuestos. En este trabajo se propone el desarrollo de un entorno de simulación basado en ROS y Gazebo que permitirá no sólo eludir cuestiones de seguridad, sino también evitar problemas logísticos que en ciertas ocasiones no se podrían llegar a asumir.**

***Index Terms*—Vehículos autónomos, entorno de simulación, control inteligente, ROS, Gazebo.**

I. Introducción

Cuando se habla de movilidad autónoma se asocia rápidamente este concepto con la movilidad urbana, y pocas veces se perciben otras áreas igual de interesantes donde esta tecnología podría ser desplegada. Estas otras áreas generalmente tratan con entornos complejos donde el desarrollo y el conocimiento humano se verían potenciados si existieran dominios seguros simulados para el depurado de fallos. Son entornos como la arqueología o la espeleología, cuyos espacios de trabajo suelen ser difíciles y peligrosos, o incluso en la carrera espacial, campo en el que, de un tiempo a esta parte, ha surgido la oportunidad de abarcar mundos inexplorados. Con todo, una de esas áreas en concreto es la movilidad turística, cuyas actividades se verían enormemente afectadas en beneficio tanto del visitante como del organismo que las realiza, impulsando la preservación de sus espacios naturales y el disfrute de los turistas. Así, esta área junto con las mencionadas anteriormente serían favorecidas por un conjunto de metodologías y herramientas capaces de generar espacios simulados, seguros y fiables que permitirían el desarrollo y depurado en las plataformas autónomas que trabajan en ellos. De esta manera, se podrían generar procesos flexibles que permitan moldear fácilmente un conjunto de entornos para ajustarse a las necesidades de la plataforma y su espacio de trabajo, evitar problemas logísticos para el transporte y la certificación de un vehículo autónomo y garantizar la robustez y fiabilidad de los sistemas, avalando así, no solo la madurez de la tecnología actual, sino también, la respuesta ante fallos de cualquier plataforma que trabaja en entornos complejos.

Para consolidar un desarrollo seguro y fiable, se propone el uso de un entorno capaz de simular modelos robóticos 3D así como físicas realistas, un aspecto importante en el control de grandes vehículos para el transporte de personas. Existen múltiples alternativas para la simulación de vehículos, unas más conocidas por la industria como CarSim, otras más centradas en el mundo académico como Carla [1] o USARSim [2]. Si bien resultan ser potentes herramientas, su elevada especificidad suele chocar contra otros entornos de desarrollo robótico, encontrando impedimentos que obligan a optar por simuladores más flexibles para generar modelos de vehículos poco comunes, y maximizar la compatibilidad con el paradigma en el que se desarrolla. Así nace Gazebo, [3] como componente para el proyecto Player [4] en 2004, hasta 2011 que pasa a ser un programa independiente convirtiéndose en uno de los simuladores multiplataforma más utilizados actualmente por su flexibilidad y sencillez. Además, Gazebo permite registrar distintos motores de físicas: ODE [5], Bullet [6], Simbody [7] y Dart [8] para ajustar aún más la simulación a las necesidades y requisitos del vehículo en cuestión.

Gazebo ha ido adquiriendo notoriedad e importancia dentro del campo de la robótica y, de un tiempo a esta parte, ha desarrollado e integrado una gran cantidad de herramientas y paradigmas para la simulación de robots convirtiéndose en una potente herramienta de desarrollo y depurado [9]. De esta manera, es posible encontrar trabajos actuales que han sabido explotar al máximo las funcionalidades del simulador: en vehículos terrestres [10] [11], para el depurado del control en robots [12] [13], o incluso para el desarrollo en vehículos no tripulados [14] [15] o la exploración espacial [16], como es el caso del rover Curiosity mandado por la Nasa a Marte. Aunque si bien es cierto que su uso comprende varias áreas de conocimiento focalizadas en distintos tipos de vehículos, hay una relativa carencia en el estado del arte por transportar el conocimiento obtenido en la simulación de estos modelos a vehículos de mayor escala como automóviles o autobuses. Siendo el área de la movilidad autónoma una de las que han adquirido mayor relevancia desde hace varios años en el campo de la robótica y automatización, resulta relevante establecer los términos que posibilitarían el desarrollo ágil y

seguro sobre vehículos autónomos. Así, existen trabajos que se centran en vehículos pequeños pero que no llegan a ser como un automóvil estándar [17] [18]; por ello, en este trabajo se pretende abordar la simulación de una plataforma móvil de grandes dimensiones, autónoma y eléctrica que aunará el conocimiento obtenido en otros trabajos similares aplicándolo a nuevos vehículos y modelos de control.

Este modelo de control beberá directamente de una herramienta integrada en Gazebo que facilita la implementación y testeo de los sistemas embarcados en sus equivalentes reales. ROS [19], Robot Operating System, es un *middleware* que proporciona muchas utilidades esenciales para el desarrollo distribuido, especialmente centrado en robótica. ROS permite el uso de estándares, técnicas y prácticas bien definidas y avaladas por la comunidad de investigación robótica. En particular, cuenta con una metodología llamado ROS Control [20]: un conjunto de herramientas que ofrecen una serie de interfaces y componentes que permiten implementar controladores tanto en simulación, como en un robot real. Así, mientras que Gazebo aporta un entorno seguro, ROS proporciona la capacidad para el testeo de los sistemas de control que se van desarrollando continuamente y que irán embarcados en la plataforma real. A continuación, se propone un entorno válido, basado en las herramientas y entornos mencionados, para el depurado en cualquier plataforma móvil.

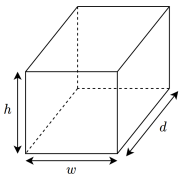
II. Desarrollo del Entorno de Simulación

El objetivo es desarrollar un entorno de simulación e integrar un modelo de vehículo completamente funcional que se acerque a la realidad lo máximo posible. Para ello sería relevante dividir esta tarea en varias cuestiones: la primera trataría la descripción del vehículo mediante las herramientas de ROS: Xacro (XML Macros): un lenguaje XML para la generación de macros que simplifican el desarrollo de código, y URDF (*Unified Robot Description Format*): un estándar para la descripción de modelos robóticos. Por otro lado, estaría la generación del mundo simulado con el que el modelo interaccionará. Finalmente, será necesario adecuar el motor de físicas para ajustarse a la dinámica de la plataforma real e integrar un controlador para generar el movimiento estable del modelo.

A. Generación del Modelo Robótico

La generación del URDF se pauta con un desarrollo modular, donde cada componente es funcional de manera independiente, por lo que se podrá crear el modelo final por piezas, con la posibilidad de adicionar nuevos elementos o exportarlos si fuese necesario. Así, se genera un árbol URDF, a través de un conjunto de *links*, unidos por medio de *joints*, que posteriormente darán el carácter funcional a la plataforma.

Primero se crea la estructura base. El cuerpo (*base_link*) es la parte central del modelo con una colisión de las mismas dimensiones que el vehículo y una matriz de inercias aproximada con (1). La inercia es una matriz cuadrada 3×3 que se puede definir directamente con un total de seis elementos debido a su naturaleza simétrica. Para simplificar esta matriz puede ser una aproximación, pero siempre será más adecuado calcular las inercias de la plataforma real para mayor verosimilitud.



$$I_h = \frac{1}{12} m \left(w^2 + d^2\right)$$
$$I_w = \frac{1}{12} m \left(d^2 + h^2\right) \quad (1)$$
$$I_d = \frac{1}{12} m \left(w^2 + h^2\right)$$

La matriz está asociada a un *link* independiente (*main_mass*) al cuerpo ya que esto ha demostrado un tratamiento más estable en Gazebo de masas elevadas [5]. Posteriormente se generan varios *joints* que unen el techo (*base_roofprint*), la masa del modelo y sus inercias y el cuerpo. Adicionalmente, y como una práctica general, se añade un *link* adicional que será el *footprint* (*base_footprint*), otro sistema de coordenadas desde el suelo. Todo esto conformaría la base del vehículo.

A continuación se añaden todos sus sensores. Se generan los modelos de los sensores de ultrasonidos, LiDAR, GPS, IMU, Cámaras, etc. que son independientes de la base. Estos sensores están formados por un *link* donde se parametrizan sus características (rangos mínimo y máximo, resolución, muestreo...), un *joint* fijo con una inercia ajustada al sensor real y el plugin de Gazebo que simula sus lecturas. Finalmente, se posicionan en el cuerpo del modelo, y se añaden una serie de condiciones que permitan cambiar sus características y su posición para producir salidas variables de datos.

En este punto solo falta implementar el sistema de ruedas, probablemente el aspecto en el que difieran la mayoría de los modelos. Generalmente, el control de un automóvil tradicional se basa en la geometría Ackermann, no obstante, para este modelo, se propone el control de un vehículo con un sistema inusual basado en un doble Ackermann, como se observa en la Fig. 1. Este sistema resulta muy útil para la maniobrabilidad del propio vehículo, y al describirlo se deben tener en cuenta múltiples conexiones entre varios elementos. Así, será necesario especificar un total de dos *links*, uno que describa la rueda de control longitudinal (*wheel_link*); con una matriz de inercias basada en un cilindro aproximada con (2). Y otro, para la rueda de control lateral (*steer_link*), sin visual y colisión, pero con la misma matriz de inercias que el anterior.

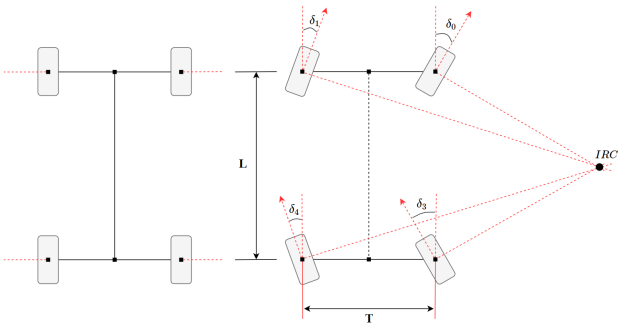
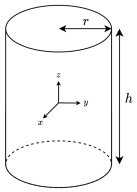


Fig. 1: Modelo doble Ackermann

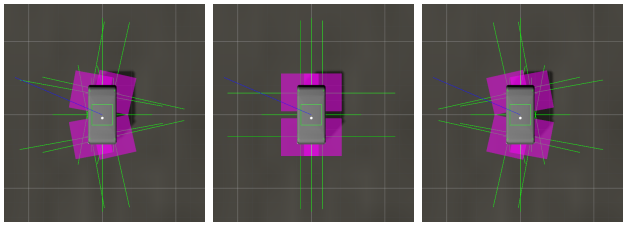


$$I_x = I_y = \frac{1}{12} m \left(3r^2 + h^2\right)$$
$$I_z = \frac{1}{2} mr^2 \quad (2)$$

Ahora se deben definir los *joints* necesarios para unir los *links* de las ruedas: uno *"continuous"*, para el avance lineal del vehículo (*wheel_joint*), y otro *"revolute"*, para la dirección (*wheel_steer_joint*). Con esto las ruedas estarían completamente modeladas, pero para aproximarse más a la realidad, se incluye una versión que añade un *link* más con inercia nula y un *joint* adicional de tipo *"prismatic"* que permite generar un sistema de suspensión (*shock_absorber*). Así, todas estas conexiones permiten respetar la estructura de árbol del estándar URDF y modelar el control longitudinal y lateral del mismo sistema de *links* simultáneamente.

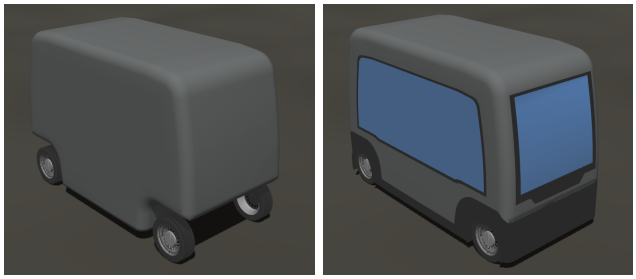
La Fig. *2* muestra la geometría final del sistema de ruedas. Las líneas verdes son los ejes de la geometría *Ackermann* y las cajas moradas, las inercias de cada *link*. Como se puede observar, los ejes se alinean en el giro para establecer un centro instantáneo de rotacion (IRC). Adicionalmente, el URDF permite establecer los *"safety controllers"* para estos ejes, que permiten ajustar los valores seguros de esfuerzo, velocidad y posición para cada tipo de vehículo.

Finalmente, las ruedas se deben instanciar en el Xacro global, junto con la base, los sensores y los parámetros de configuración. Esto conformaría el modelo final. No obstante, para dar realismo a la simulación se añaden los elementos *"visual"* al URDF, en forma de mallas 3D para tener una visual del modelo, como se puede ver en la Fig. 3.



(a) Giro derecha (b) Ruedas centradas (c) Giro izquierda

Fig. 2: Geometría ruedas del modelo



(a) Modelo 3D base (b) Modelo 3D final

Fig. 3: Modelos 3D del vehículo

En la Fig. 4 se muestra el árbol URDF del modelo final.

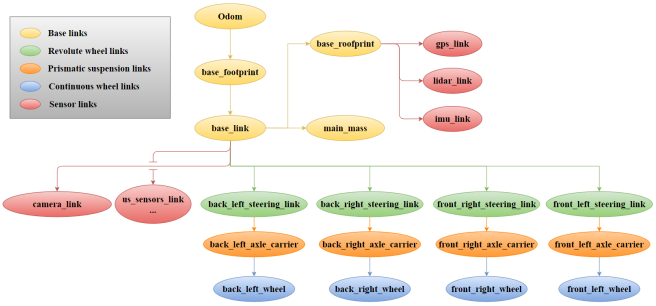
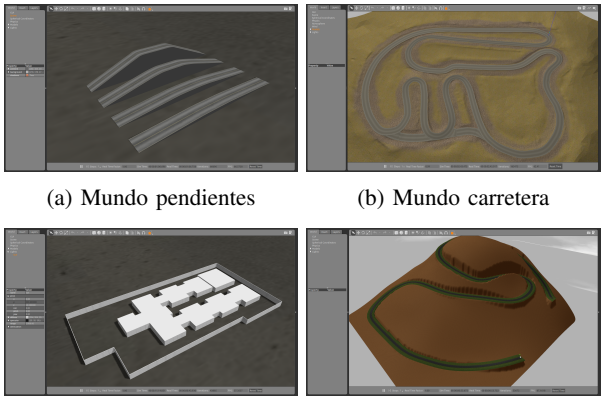


Fig. 4: Árbol URDF del modelo

B. Generación de los Mundos

Los mundos son extensiones de terreno útiles para validar los requisitos formalizados durante el desarrollo del sistema de control de la plataforma. Estos se definen en ficheros SDF de extensión *.world*, y son importantes en tanto en cuanto son los elementos con los que el modelo va a interactuar, por lo que resulta interesante saber la metodología para generarlos. En la Fig. 5 se puede observar algunos de los mundos generados.



(a) Mundo pendientes (b) Mundo carretera

(c) Mundo urbano (d) Mundo montaña

Fig. 5: Mundos generados en Gazebo

El mundo se define como una entidad SDF independiente y posee una serie de parámetros que determinan el comportamiento de los modelos que se hallan en él. Entre ellos destacamos efectos atmosféricos, fuentes de luz y el motor de físicas, que se tratará en la siguiente sección. Además, a cada mundo se deben añadir los modelos del entorno (carretera, suelo, obstáculos, etc.) y adecuar sus parámetros físicos, que definirán el comportamiento del vehículo al contacto con ellos. Así, el mundo vacío se irá poblando con modelos adicionales con los que se consumará el entorno en el que trabajará el vehículo. Siguiendo esta metodología se llega a generar un conjunto de mundos para el depurado de distintos aspectos del sistema: control lateral y longitudinal, detección de la carretera y obstáculos, validación del sistema de control etc.

C. Funcionamiento y Ajustes de ODE

Gazebo permite ajustar el motor de físicas para adecuarse al entorno con el que se trabaja. Tanto es así que el simulador integra los motores ODE, Bullet, Simbody y Dart. Simbody y

Dart se inclinan más hacia la robótica de articulaciones; en el caso de Bullet, actualmente está más enfocado en animaciones y juegos, siendo ODE mucho más manejable desde la API de Gazebo al ser su motor "nativo". Para este proyecto en concreto se ha elegido ODE como motor de físicas por la facilidad de integración con la plataforma de desarrollo (ROS).

Uno de los aspectos fundamentales a tratar en un motor de físicas es el modelo de fricción. ODE genera este modelo a través de un conjunto de variables que se definen globalmente en el mundo e individualmente en cada *link*. El mundo (*dWorld*) almacena cuerpos (*dBody*) y calcula los esfuerzos sufridos y su posición produciendo el siguiente estado de la simulación. Con todo, cuando dos cuerpos están suficientemente juntos, se hace una llamada a la API de ODE para generar los *"contact joints"* (*dJoint*), advirtiendo a Gazebo que dos cuerpos están en contacto y producen fuerzas entre ellos, instanciando, como resultado, un espacio de colisiones (*dSpace*). Para ello, ODE posee diferentes modelos de fricción: piramidal, cono o caja, y utiliza por defecto el piramidal, como aproximación al de cono. El modelo de cono se basa en la fricción de Coulomb (μ) y relaciona fuerzas normales (F_N) y tangenciales (F_T) con (3) en un punto de contacto. A continuación se proponen cambios para ajustar el modelo.

$$F_T \leq \mu \cdot |F_N| \tag{3}$$

1) Masas elevadas: Una de las limitaciones de Gazebo es la unión de *links* con masas muy dispares. Como se modela un vehículo es probable acabar en esta situación, en la que ODE tiende a ser inestable. Para evitar esto se suele dividir la masa en distintas colisiones que posteriormente se agrupan, o separarla en uno o varios *links* virtuales y así disminuir la diferencia de masas entre cuerpos.

2) Ajuste del mundo: El mundo se puede adaptar con tres parámetros fundamentales: CFM (*Constraint Force Mixing*), ERP (*Error Reduction Parameter*) e *iters* (número de iteraciones), que cuanto mayor más precisión en deterioro del desempeño de Gazebo. Estos parámetros se tratarán más detenidamente en la sección II-C3 ya que son más relevantes en cada *link* individual del modelo. Por otro lado, también se pueden ajustar los valores del motor de físicas: *contact_max_correcting_vel*, permitiendo un número determinado de velocidades para resolver los contactos de una manera más suave, y *contact_surface_layer* que determina la distancia de penetración al contacto. Valores más pequeños de este último suelen ayudar a disminuir las sacudidas en contactos defectuosos o mal calculados. Finalmente, hay casos en los que la carga de las colisiones supone un coste computacional alto. Esto se debe a que ODE calcula en cada ciclo de simulación las colisiones de cada malla y, en aquellas que son complejas, se experimenta una caída del rendimiento. Para simular grandes entornos afectando lo mínimo posible al rendimiento en simulación, existen varias posibilidades:

- **Colisiones con formas primitivas.** Útil para mallas pequeñas, sin formas elaboradas, como una carretera aproximada a un plano. Sin embargo, en mallas complejas es poco eficaz al no cubrir toda la superficie de la misma.
- **Simplificación de la malla.** Es posible utilizar herramientas como MeshLab para limpiar la malla, detectar inconsistencias y disminuir el número de elementos. Así, mallas simplificadas generarían colisiones más eficientes.
- **Grupos de colisión.** ODE permite definir colisiones como unión de otras más pequeñas. Así, es posible definir distintas colisiones en una malla grande y hacer que el motor sólo calcule contactos para aquellas que interesen con (*collide_without_contact*). Esto determina una colisión sin contacto y permite no saturar Gazebo, al mismo tiempo que siguen siendo útiles para los sensores.

3) Ruedas: Las características físicas de la rueda afectan al contacto con el suelo, así como al comportamiento de sus controladores. Para modelar los contactos, ODE genera un conjunto de *"contact joints"* sobre los que hace cómputos. Si valores de masa, esfuerzo etc. aplicados superan ciertos límites o son muy dispares, el simulador tiende a ser inestable. Las ecuaciones generadas en este caso carecen de soluciones adecuadas, y dan lugar a esfuerzos incorrectos al aplicar fuerzas en los *joints* del modelo. Así, es posible que exista un problema de convergencia para el algoritmo de Gauss-Seidel Proyectado de ODE si el número de condición de la matriz de restricción $J\left(M^{-1}\right)J^T$ es elevado, haciendo que ésta sea mal condicionada (*ill-conditioned*). Como solución, se puede:

- **Cambiar colisión.** La colisión de la rueda se puede modelar con un cilindro, esto quizás genere puntos de contacto óptimos insuficientes ya que ODE elimina los que considera poco adecuados. Si no hay un conjunto apropiado, es posible que haya perturbaciones al aplicar fuerzas erróneas. Así, es posible optar por múltiples colisiones cilíndricas, que ofrecen más contactos con el suelo y un conjunto mayor de puntos, o, por una colisión esférica que produce un contacto mucho más estable y sin tantas perturbaciones en cada paso del motor.
- **Ajustar colisión y fricción en ODE.** Se pueden ajustar los parámetros de contacto *kp, kd, mu y mu2*. *kp* es el coeficiente de rigidez al contacto: resistencia a las deformaciones elásticas, y *kd* el coeficiente de amortiguación: velocidad a la que las oscilaciones en un sistema se estabilizan. Estos son traducidos a los valores nativos de ODE: ERP y CFM, mediante (4). Por otro lado, *mu* es el coeficiente de fricción de Coulomb para la primera dirección de fricción y *mu2* es el coeficientes de fricción para la dirección perpendicular a la de *mu*. Finalmente, h es lo que se conoce como *stepSize* y controla el número de iteraciones a realizar por cada paso del motor.

$$ERP = \frac{h\,kp}{(h\,kp + kd)} \quad CFM = \frac{1}{(h\,kp + kd)} \tag{4}$$

El ERP corrige el error en los *joints*. Cada *joint* une dos cuerpos por una articulación y asocia unas *poses* relativas. Cuando se mueve el *joint*, es posible que los cuerpos acaben en una *pose* en la que las restricciones establecidas

por ODE no se cumplen. Como en tiempo de simulación todo se calcula de manera secuencial, los errores se van acumulando y ODE aplica una pequeña fuerza para alinear los cuerpos y posicionarlos correctamente.
El CFM cambia la rigidez de los *joints*, controlando las restricciones "suaves" y "duras", simulando un *joint* flexible o uno rígido, respectivamente. Las restricciones "duras" son inviolables, pero en ODE no ocurre exactamente esto al introducir el CFM para contemplar la posibilidad de violar una restricción. El comportamiento de los *joints* se define mediante la ecuación de restricción (5), donde v es el vector de velocidades del cuerpo, J es la matriz jacobiana con tantas filas como grados de libertad del *joint* y c el *Right hand side vector*.

$$J\,v = c \tag{5}$$

En cada iteración se busca calcular un vector en (6) que, introducido en la ecuación, mantenga las fuerzas aplicadas para respetar las restricciones establecidas.

$$\begin{aligned} F_c &= J^T\lambda \\ J\,v &= c + CFM\,\lambda \end{aligned} \tag{6}$$

El parámetro mezcla las fuerzas calculadas con las restricciones que las producen. Así, resolviendo para λ se obtiene la ecuación (7).

$$\left(JM^{-1}J^T + \frac{CFM}{h}\right)\lambda = \frac{c}{h} \tag{7}$$

En resolución, CFM es una matriz diagonal que posibilita violar las restricciones originales del sistema de manera proporcional a su valor. Un valor positivo permitiría alejarse de posibles singularidades en simulación. Así, con los siguientes valores propuestos:

- $mu = mu2 = 0.9$
- $kd = 1 \times 10^9$, $kp = 5 \times 10^6$
- $h = 1 \times 10^{-3}$

Se calcula un $ERP \simeq 0.17$ y $CFM \simeq 1.7e{-}07$, valores con los que ODE opera de manera estable con el modelo.

- **Facilitar contactos.** En ocasiones, la rigidez del modelo no permite que todas las ruedas entren en contacto con el suelo. Para asegurar el contacto, independientemente de las irregularidades del terreno, se generan los *joints "prismatic"* para modelar la suspensión, y se ajustan los valores de *implicitSpringDamper*, *springStiffness* y *springReference*.
  - $implicitSpringDamper = true$
  - $springStiffness = 0.25$
  - $springReference = 1 \times 10^{-4}$

III. Desarrollo del Controlador

En este punto, el modelo es un conjunto de *links* estático, por lo que se deben generar los actuadores para controlar su movimiento. Los actuadores en URDF se definen con *Transmissions*, elementos que realizan un mapeo entre un *joint* y un actuador. Conectan el *joint* a ROS Control y advierten a Gazebo sobre cómo debe comportarse. Dado que el modelo posee una geometría doble Ackermann, se añaden dos por cada rueda: un controlador de velocidad (*VelocityJointInterface*), y otro de posición para el giro (*PositionJointInterface*). Asimismo, se implementan dos nodos de ROS: *Model CAN Sim*, que simula las señales del Bus CAN, y *Model Controller*, que toma como entradas el estado de los *joints* y los mensajes Ackermann del sistema de control del vehículo y produce las señales que se envían a cada actuador, generando los esfuerzos correspondientes para mover la plataforma. Además, se crean los archivos de configuración para Gazebo, ODE y ROS Control, donde se instancian las interfaces de control. En la Fig. 6 se muestra el entorno propuesto para el control. Nótese la encapsulación de la arquitectura de control de la plataforma real, por lo que el comportamiento en simulación sería muy similar al que se pudiera ver en la realidad.

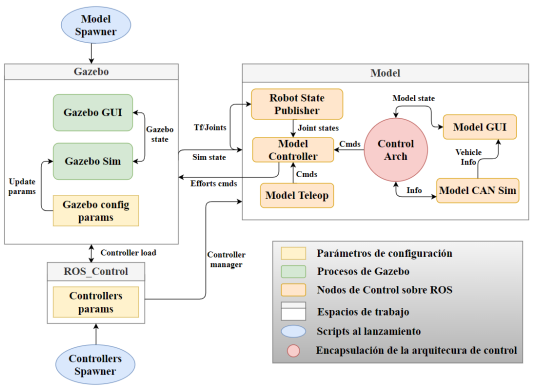
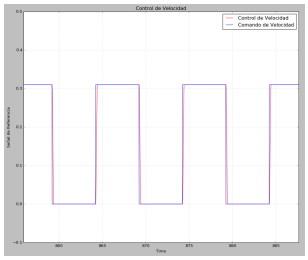


Fig. 6: Entorno para el control

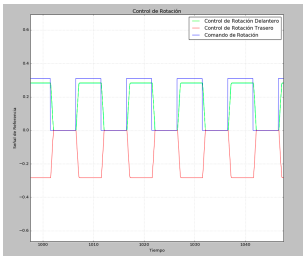
IV. Resultados

La Fig. 7 muestra el control del modelo, estable longitudinal y lateralmente, con un desfase al ser el movimiento no instantáneo por la dinámica de la plataforma, y un giro menor que la señal de control al contar con un limitador de dirección. La respuesta ante el valor de consigna es adecuada en llano y los controladores son capaces de actuar venciendo la masa de la plataforma y el rozamiento con el suelo. Con todo, debido a las limitaciones de URDF, los elementos superiores del árbol (control lateral) introducen perturbaciones en los inferiores (control longitudinal), Fig. 7c. Si bien el sistema tiende al equilibrio, será necesario ajustar la velocidad de giro y la aceleración de la plataforma para que el control lateral no desestabilice al control longitudinal cuando intenta mantener las ruedas en la posición requerida, Fig. 7d.

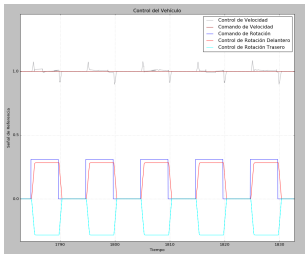
Finalmente, la Fig. 8 muestra el control en pendiente: la Fig. 8a representa las perturbaciones a la entrada de una pendiente, siendo habituales al producirse el contacto de las ruedas con ella, momento en el que actúa el controlador de giro manteniendo la posición requerida por el valor consigna de giro; y la Fig. 8b muestra la parada en pendiente y el error mínimo de velocidad para el que la plataforma cae por su peso. Esto último dependerá de su inclinación.



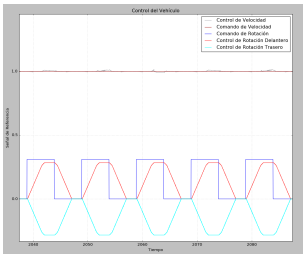
(a) Control longitudinal



(b) Control lateral

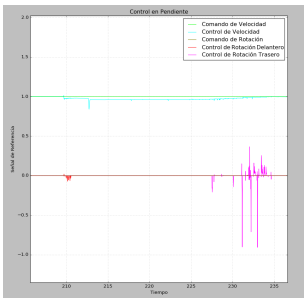


(c) Control completo inestable

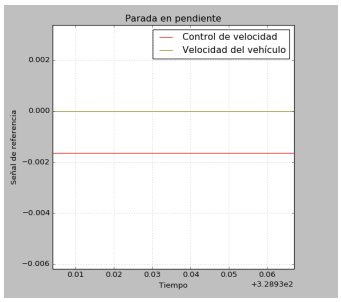


(d) Control completo estable

Fig. 7: Control del modelo



(a) Control a la entrada



(b) Control en parada

Fig. 8: Control en pendiente

V. Conclusión

En este trabajo se ha generado un entorno seguro y fiable para el depurado de un vehículo autónomo. Se ha adecuado el simulador Gazebo y el motor de físicas ODE para modelar una plataforma móvil de manera realista. Asimismo, se ha producido un entorno de control basado en ROS, lo que ha permitido añadir las características funcionales al modelo, demostrando un comportamiento estable ajustado a la realidad. Con todo, la mecánica a bajo nivel de la plataforma simulada no deja de ser una aproximación a la realidad y resulta inevitable que algunos aspectos de la misma carezcan de realismo. Por este motivo, se debe seguir trabajando en este ámbito para mejorar las interacciones entre componentes y las prestaciones de sus herramientas, intentando generar metodologías que aseguren mejores aproximaciones, lo que en un futuro podría permitir la detección precoz de fallos o comportamientos erráticos antes de que sucedan en la realidad.

Agradecimientos

Investigación avalada por el gobierno de España a través de los proyectos de la CICYT (PID2019-104793RB-C31 y RTI2018-096036-B-C21) y de la Comunidad de Madrid a través de SEGVAUTO-4.0-CM (P2018/EMT-4362).

Referencias

[1] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An open urban driving simulator," in *Proceedings of the 1st Annual Conference on Robot Learning*, 2017, pp. 1–16.

[2] S. Carpin, M. Lewis, J. Wang, S. Balakirsky, and C. Scrapper, "Usarsim: a robot simulator for research and education," in *Proceedings of International Conference on Robotics and Automation*, 2007, pp. 1400–1405.

[3] N. Koenig and A. Howard, "Design and use paradigms for gazebo, an open-source multi-robot simulator," in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566)*, vol. 3. IEEE, 2004, pp. 2149–2154.

[4] B. Gerkey, R. Vaughan, and A. Howard, "The player/stage project: Tools for multi-robot and distributed sensor systems," *Proceedings of the International Conference on Advanced Robotics*, 08 2003.

[5] R. Smith *et al.*, "Open dynamics engine," *http://ode.org/*, 2005.

[6] E. Coumans, "Bullet physics engine," *Open Source Software: http://bulletphysics. org*, vol. 1, no. 3, p. 84, 2010.

[7] M. Sherman, A. Seth, and S. Delp, "Simbody: Multibody dynamics for biomedical research," *Procedia IUTAM*, vol. 2, pp. 241–261, 12 2011.

[8] J. Lee, M. Grey, S. Ha, T. Kunz, S. Jain, Y. Ye, S. Srinivasa, M. Stilman, and C. Liu, "Dart: Dynamic animation and robotics toolkit," *The Journal of Open Source Software*, vol. 3, p. 500, 02 2018.

[9] C. C. Cuevas Castañeda, "Ros-gazebo. una valiosa herramienta de vanguardia para el desarrollo de la robótica," *Publicaciones e Investigación*, vol. 10, pp. 145–160, mar. 2016.

[10] K. Takaya, T. Asai, V. Kroumov, and F. Smarandache, "Simulation environment for mobile robots testing using ros and gazebo," in *2016 20th International Conference on System Theory, Control and Computing (ICSTCC)*, 2016, pp. 96–101.

[11] M. Sokolov, R. Lavrenov, A. Gabdullin, I. Afanasyev, and E. Magid, "3d modelling and simulation of a crawler robot in ros/gazebo," in *Proceedings of the 4th International Conference on Control, Mechatronics and Automation*, ser. ICCMA '16. New York, NY, USA: Association for Computing Machinery, 2016, p. 61–65.

[12] W. Yao, W. Dai, J. Xiao, H. Lu, and Z. Zheng, "A simulation system based on ros and gazebo for robocup middle size league," in *2015 IEEE International Conference on Robotics and Biomimetics*, 2015, pp. 54–59.

[13] W. Qian, Z. Xia, J. Xiong, Y. Gan, Y. Guo, S. Weng, H. Deng, Y. Hu, and J. Zhang, "Manipulation task simulation using ros and gazebo," in *Proceedings of International Conference on Robotics and Biomimetics*, 2014, pp. 2594–2598.

[14] M. M. M. Manhães, S. A. Scherer, M. Voss, L. R. Douat, and T. Rauschenbach, "Uuv simulator: A gazebo-based package for underwater intervention and multi-robot simulation," in *OCEANS 2016 MTS/IEEE Monterey*, 2016, pp. 1–8.

[15] J. Meyer, A. Sendobry, S. Kohlbrecher, U. Klingauf, and O. von Stryk, "Comprehensive simulation of quadrotor uavs using ros and gazebo," in *Simulation, Modeling, and Programming for Autonomous Robots*, I. Noda, N. Ando, D. Brugali, and J. J. Kuffner, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 400–411.

[16] R. C. Cardoso, M. Farrell, M. Luckcuck, A. Ferrando, and M. Fisher, "Heterogeneous verification of an autonomous curiosity rover," in *NASA Formal Methods*, R. Lee, S. Jha, A. Mavridou, and D. Giannakopoulou, Eds. Cham: Springer International Publishing, 2020, pp. 353–360.

[17] Shimchik, Ilya, Sagitov, Artur, Afanasyev, Ilya, Matsuno, Fumitoshi, and Magid, Evgeni, "Golf cart prototype development and navigation simulation using ros and gazebo," *MATEC Web Conf.*, vol. 75, 05 2016.

[18] K. Shabalina, A. Sagitov, K.-L. Su, K.-H. Hsia, and E. Magid, "Avrora unior car-like robot in gazebo environment," *Proceedings of International Conference on Artificial Life and Robotics*, vol. 24, pp. 116–119, 01 2019.

[19] O. S. R. Foundation. (2018, 01) Is ros for me? OSRF. [Online]. Available: http://www.ros.org/is-ros-for-me/

[20] S. Chitta, E. Marder-Eppstein, W. Meeussen, V. Pradeep, A. Tsouroukdissian, J. Bohren, D. Coleman, B. Magyar, G. Raiola, M. Lüdtke, and E. Perdomo, "ros_control: A generic and simple control framework for ros," *The Journal of Open Source Software*, vol. 2, p. 456, 12 2017.

Detección y modelado de la carretera mediante el análisis morfológico de nubes de puntos

Alejandro Ruiz-de-la-Cuadra*, David Yagüe-Cuevas†, Pablo Marín-Plaza‡ and José María-Armingol§
Laboratorio de Sistemas Inteligentes, Universidad Carlos III de Madrid
Leganés, Madrid
Email: *aruizla@inf.uc3m.es, †dayaguec@inf.uc3m.es, ‡pamarinp@ing.uc3m.es, §armingol@ing.uc3m.es

***Abstract*—La detección de la carretera es un elemento principal para el mapeo y la navegación de vehículos autónomos, así como un papel importante en el campo de la seguridad vial y el mantenimiento de la carretera [1]. En las últimas dos décadas la mayoría de los algoritmos se han centrado en los entornos urbanos, los cuales sufren principalmente de la oclusión de otros vehículos y/u otros elementos. Sin embargo, se caracterizan por ser altamente estructurados con unos límites definidos por marcas viales y bordillos uniformes [2]. En este trabajo se ha diseñado un modelo probabilístico que permite detectar cambios significativos en el perfil de carreteras secundarias sin marcas viales ni bordillos para identificar los límites que definen el espacio navegable. Los resultados experimentales muestran una segmentación efectiva de la carretera en distintos escenarios que permiten hacer un seguimiento reactivo del centro de la calzada.**

***Index Terms*—LiDAR, nubes de puntos, vehículos autónomos, detección y modelado de la carretera, MLS.**

I. Introducción

La principal motivación detrás de la integración de los vehículos autónomos es incrementar el flujo y la capacidad de tráfico debido a la mejora en los tiempos de respuesta, reducir el consumo y la polución gracias a la previsión en la conducción, y alcanzar una menor tasa de accidentes gracias a los sistemas de evasión de colisiones.

Estos vehículos equipan una combinación de sensores de percepción: cámara, RADAR, LiDAR, sistema de navegación global por satélite (GNSS) y unidad de medición inercial (IMU), para hacer posible la navegación autónoma a través de carreteras urbanas y rurales [3]. Proyectos destacados como los primeros ganadores del DARPA, Stanley [4] y Boss [5] en carreteras no estructuradas y urbanas respectivamente, destacan la importancia de los sistemas de la detección de la carretera. En el caso de Stanley, la detección de los límites de la carretera permite al vehículo conducir centrado a la carretera para evitar la mayoría de los obstáculos. Mientras el caso de BOSS permite modelar la forma de la carretera permitiendo navegar por el pavimento y por todo terreno.

Los algoritmos de detección de carretera definen el espacio navegable y los límites, dotando de robustez tanto a la planificación de movimiento como al control. El estudio del análisis de la carretera no se ha enfocado únicamente en el mapeo o control de vehículos no tripulados, sino que también se aplica al campo de seguridad vial y mantenimiento de la misma [6] [7], entre otras aplicaciones.

Estos algoritmos llevan estando presente los últimos 20 años, con la primera mención a la segmentación de carretera en 2001 [2], comenzando a usar cámaras y LiDAR 2D para el mapeo de la carretera. No obstante, en los últimos 15 años la detección de la carretera se ha divido en dos vertientes.

Por una parte, los algoritmos basado en visón por computación se aplican en escenarios estructurados y con alto contraste, que permiten aplicar algoritmos como la transformada de Hough o el operador del Sobel para extraer características que concuerden con los límites. Estos límites tienen la ventaja de un bajo coste y las imágenes proporcionan gran información contextual. Por el contrario, están limitados a las condiciones ambientales, elementos dinámicos, sombras, etc., provocando que estos sistemas por si sólos obtengan un alto número de falsos positivos en entornos monocromáticos o urbanos [8]. Algunos métodos destacados de visión por computación son [9] que combinan la señal de GPS y las características extraídas de la carretera para obtener la posición del vehículo, [10] un sistema de redes neurales que analizan los cambios en escala de grises para detectar intersecciones en carreteras y [11] analiza agrupaciones de píxeles para encontrar posibles candidatos usando el algoritmo de K-vecinos.

Por otro lado los sensores LiDAR proporcionan una alta precisión y resolución angular, manteniendo un amplio campo de visión frente al RADAR. Estos sensores trabajan independientemente de las condiciones de luz y devuelven una medida precisa de la distancia directamente. Es uno de los sensores más populares en robots móviles [2], obteniendo precisos resultados para el mapeado móvil, aunque con el inconveniente de su alto coste comparado con la cámaras. [12] y [13] son enfoques parecidos para clasificar puntos en entornos urbanos utilizando cámaras en estéreo y LiDAR 3D respectivamente.

Los proyectos de escaneo móvil con láser (MLS) comenzaron empleando sensores 2D y métodos de localización para acumular los puntos en un modelo tridimensional, normalmente a partir del filtro de Kalman extendido [14] [8]. Gracias a las mejoras en los procesadores y la paralelización de procesamiento con gráficas dedicadas, ya no existe inconveniente para trabajar automáticamente con varios sistemas de percepción de múltiples planos. Empresas como Uber o Google han elegido los sensores LiDAR 3D para captar el entorno de sus vehículos autónomos. Sin embargo, es necesario habitualmente un paso previo que filtre los datos y asegurar así la premisa de ejecución en tiempo real. Algunos

filtros destacados son eliminar puntos que no pertenezcan al terreno [15] [14] o reducir el espacio 3D a una imagen de bits 2D [16] o un filtro por altura [17].

El enfoque más recurrente, denominado moving window operator, para la búsqueda y detección de la carretera en nubes de puntos consiste en analizar la nube por segmentos o anillos y clasificar cada punto empleando una ventana de tamaño fijo [18] o variable [15] [14] que compare los puntos vecinos para encontrar patrones o características morfológicas [18] [13] [8] [2] [15] [19]. Este enfoque se emplea a menudo en entornos urbanos ya que categoriza la geometría de los bordillos que delimitan el espacio navegable [18] [19]. Más métodos que buscan este tipo de patrones en las vías urbanas son [16] o [13] empleando una función sigmoide para mejorar la precisión. El análisis entre vecinos no siempre viene dado por puntos ordenados según se capturan por el sensor, otras variantes conectan los puntos por grafos [1], puntos más cercanos según una triangulación Delaunay [17].

El mayor problema de estos sistemas [18] [13] [12] [2] es que analizan un entorno urbano con la premisa que la mayoría del pavimento tiene pocas variaciones de la pendiente y buscan cambios de altura sin tener en cuenta tramos peraltados o pendientes. En los últimos 5 años se ha empleado la base de datos KITTI como un estándar comparación [20], los sistemas de visión por computación emplean redes neuronales convolucionales multicapa para obtener un rendimiento similar al estado del arte [21] [22] y aquellos con mejor rendimiento [23] [24] fusionan visión y LiDAR para obtener lo mejor de los dos mundos [25] [26].

El sistema más destacado entre la bibliografía actual es la detección de límites de calzada para entorno urbano usando LiDAR 3D analizando cambios del gradiente, con un preprocesamiento de los puntos que facilita la detección de obstáculos gracias a la extracción del suelo. Los ensayos muestran resultados muy favorables, con una precisión por encima del 93% y tiempos de procesamiento por cada captura de 36.5 ms. Este sistema completa los objetivos para este proyecto, desde el preprocesamiento de la nube para garantizar el procesamiento en tiempo real, la extracción de los límites usando un método basado en el gradiente para detectar características con un tamaño de ventana adaptativo a la hora de analizar los puntos de la nube, y finalmente, la regresión y modelado de la función que representa el perfil de la calzada [15].

En este artículo se usa un conjunto de LiDAR 3D para detectar los límites de la carretera y modelar el espacio viable para seguir el centro de la calzada mediante un control reactivo sin tener que depender de sistemas de localización como el GPS. El empleo de los LiDAR permiten hacer un análisis morfológico del terreno para extraer características que determinen la probabilidad de los puntos a pertenecer a la carretera dado una referencia para entornos de bajo contraste y sin marcas que delimiten el contorno de la carretera. El desarrollo de este trabajo se ha estructurado de la siguiente manera. La sección I contiene una breve introducción de la problemática de la segmentación de la carretera y una visión general del estado del arte en este área. La sección II detalla el funcionamiento de la detección y modelado de la carretera y profundiza en el algoritmo de búsqueda de los límites. La sección III muestra los ensayos realizados en las pistas de pruebas utilizando una configuración de tres LiDAR para maximizar la percepción. Finalmente, la sección IV presenta las conclusiones y la futuras líneas de trabajo.

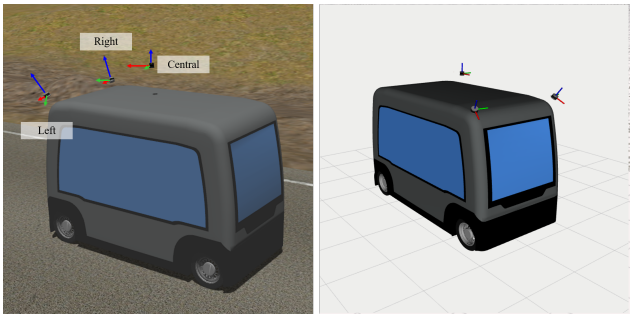


Fig. 1. Modelo del vehículo autónomo con los sensores laterales y central. Modelo en el simulador Gazebo (izquierda) y visualización Rviz (derecha)

II. Metodología

A. *Método propuesto*

El procesamiento de los datos empleado para la detección de la carretera sigue el siguiente orden. Primero, se procesa por separado cada una de las nubes de puntos obtenidas y se transforma para extraer los límites. A continuación, se segmenta cada nube en anillos para realizar de forma secuencial la búsqueda de los límites. Esta búsqueda vendrá condicionada con la información obtenida tanto de la nube anterior como del anillo previo. Finalmente, con los límites obtenidos de cada nube se realiza un ajuste por mínimos cuadrados para modelar la carretera.

B. *Procesamiento de los datos*

La adquisición de los datos provienen de tres fuentes distintas, dos LiDAR laterales y un LiDAR central (ver Fig. 1). Los puntos se toman usando tres variables: rango (ρ), la cuenta del codificador permite obtener fácilmente el acimut (θ) y el ángulo de altitud por anillo (ϕ). Se puede transforma las coordenadas esféricas de cada punto en coordenadas cartesianas usando (1) para el LiDAR central y (2) para el lateral.

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} \rho\cos(\theta)\cos(\phi) \\ -\rho\sin(\theta)\cos(\phi) \\ \rho\sin(\phi) \end{bmatrix} \quad (1)$$

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} \rho\sin(\theta)\cos(\phi) \\ \rho\cos(\theta)\cos(\phi) \\ \rho\sin(\phi) \end{bmatrix} \quad (2)$$

A continuación, se reorganizan para ser procesados por anillos. La extracción de los anillos depende de cómo cada LiDAR capte los puntos, el sensor central toma el punto j del anillo i por ángulo de elevación; mientras que los laterales se toman por el acimut. Finalmente, se ordenan los anillos en sentido horario desde π hasta $-\pi$. La posición inicial varía

de acimut ($\theta_{i,0}$) pero no es relevante ya que la búsqueda es cíclica y no lineal sobre el vector de puntos.

Para mejorar la capacidad de ejecución en tiempo real se filtran los anillos usando un filtro condicional que elimina aquellos puntos fuera de la zona más cercana del vehículo. La selección de esta zona dependerá de la disposición del sensor en el vehículo. En la Fig. 2 se aprecia como gracias a este método se reducen los puntos a cerca del 20% de la nube original.

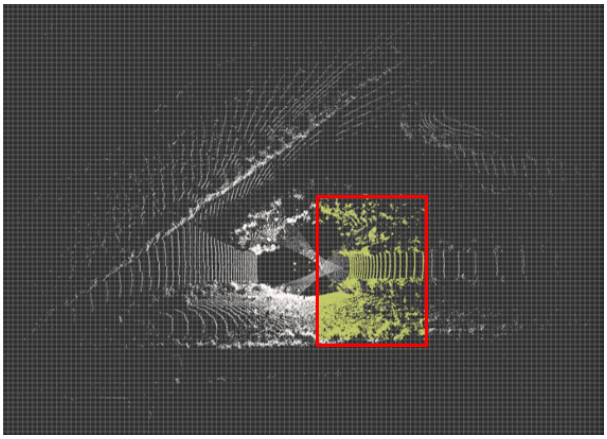


Fig. 2. Filtro condicional para LiDAR central de 64 planos.

C. Detección de los límites de la carretera (RED)

Para seleccionar los límites de la carretera (Road Edge Detection, RED) se usa un algoritmo basado en el operador de vecindad (window operator). Este operador toma las características geométricas de cada punto: pendiente, uniformidad y estimación del límite, para asignar una puntuación a los puntos. La búsqueda es bidireccional, comenzando desde la posición $s \in [1, n_i]$ hasta encontrar los límites izquierdo $p_{i,L}$ y derecho $p_{i,R}$.

El algoritmo de búsqueda varía levemente cuando se usa el ángulo frontal y el posterior (${\theta_s}^F$, ${\theta_s}^B$). Estos ángulos se instancian a 0 y pi respectivamente, partiendo de la hipótesis que el vehículo se encuentra orientado hacia la carretera y se irá actualizando la referencia a medida que se desplaza. Antes de realizar la búsqueda, independientemente del ángulo, ha de asignarse a cada punto la probabilidad de pertenecer a la carretera o point score ($P_r(p)$). Después, se seleccionan los límites usando un umbral (ε_b).

1) Puntuación de un anillo: Cuando se procesa un anillo los puntos tienen que tener asociada una puntuación. Cada puntuación se calcula usando una función sigmoide (3) que fusiona las características morfológicas de los puntos: el gradiente (4), la desviación media (5) y la estimación (6). Los coeficientes β han sido seleccionados usando el método de mínimo cuadrados para optimizar la detección de límites dada la configuración y el entorno de operación.

$$P_r(p_{i,j}) = \frac{1}{1 + e^{\beta_0 * (\beta_1 + gradient + sigma + estimation)}} \quad (3)$$

$$gradient = \beta_2 + \beta_3 * \nabla_{i,j} + \beta_4 * {\nabla_{i,j}}^2 + \beta_5 * {\nabla_{i,j}}^3 \quad (4)$$

$$sigma = \beta_6 + \beta_7 * \sigma_{i,j} + \beta_8 * \sqrt{\sigma_{i,j}} \quad (5)$$

$$estimation = \beta_9 + \beta_{10} * \hat{\Delta}_{i,j} \quad (6)$$

2) Operador de vecindad: Es el conjunto de puntos adyacentes al punto j para calcular $\nabla_{i,j}$ y $\sigma_{i,j}$, y por tanto asignar una puntuación. Para aquellos puntos ubicados a la izquierda del punto de referencia, los vecinos se seleccionan en sentido horario (ver Fig. 3). Por el contrario, los vecinos se seleccionan en sentido contra horario. Para determinar si un punto se ubica a la izquierda o derecha del punto de referencia, se emplea la resta normalizada de acimut $\Delta\theta_{i,j}$ (1), la cual soluciona el problema aquellos ángulos pertenecientes a distintos sectores.

Algorithm 1 Resta normalizada de acimut

Input: $\theta_{i,j}$, θ_s
Output: $\Delta\theta_{i,j}$
Initialisation : $\Delta = \theta_s - \theta_{i,j}$
if $-\frac{3\pi}{2} > \Delta$ **then**
  return $remainder(\Delta, \pi)$
else if $\Delta < -\pi$ **then**
  return $\pi + \mod(\Delta, \pi)$
else if $\Delta \leq \pi$ **then**
  return Δ
else
  return $remainder(\Delta, \pi)$
end if

A continuación, el número de vecinos $w_{i,s}$ que forman el operador es el número teórico de puntos consecutivos en una sección del escaneo láser w_s en metros (7). El tamaño del operador depende de la altura de los límites en el entorno y de la precisión del LiDAR, valores $w_s = [0.2, 0.3]$ ofrecen similares resultados para esta configuración.

$$w_{i,s} = \left\lceil \frac{w_s}{\rho_{i,j} * \gamma} \right\rceil \quad (7)$$

Finalmente, los vecinos se seleccionan usando el ángulo de búsqueda más cercano y su resta normalizada $\Delta\theta_{i,j}$. Cuando $\Delta\theta_{i,j}$ es positiva, entonces se seleccionan puntos desde la posición j hasta $(j + w_{i,s}) \bmod n_i$, y desde $(j - w_{i,s}) \bmod n_i$ hasta j en caso contrario.

$$\Delta\theta_{i,j} = \min\left\{\left|\Delta{\theta_{i,j}}^F\right|, \left|\Delta{\theta_{i,j}}^F\right|\right\} \quad (8)$$

3) Búsqueda de los límites: Una vez se hayan calculado todos los puntos, el siguiente paso es encontrar los límites en sentido horario ${p_{i,R}}^F$, ${p_{i,L}}^B$ y/o en sentido contra horario (${p_{i,L}}^F$, ${p_{i,R}}^B$).

Comenzando siempre por $j = s$, el algoritmo de búsqueda (2) comprueba por cada punto en secuencia su puntuación hasta que $P_r(p_{i,j})$ supere ε_b. Este algoritmo devuelve los límites izquierdo $P_b(p_{i,R})$ y derecho $P_b(p_{i,L})$ con su puntuación actualizada.

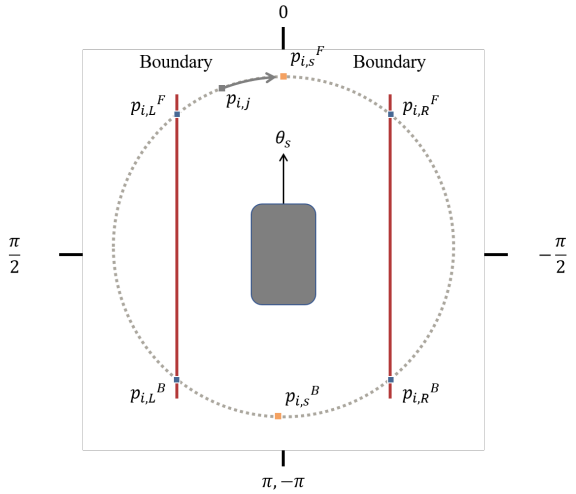


Fig. 3. Diagrama del operador de vecindad.

La puntuación del anillo $P_b(r)$ se obtiene usando las puntuaciones de todos los puntos dentro de la carretera y los límites. Primero, el algoritmo (2) guarda todos los valores mínimos de los puntos $P_r(p_{i,min})$ entre los límites izquierdo y derecho. Después calcula la media de las puntuaciones para obtener una comparativa más realista con las puntuaciones de los límites. Finalmente, el límite con mayor puntuación se elige para substraer la puntuación media siguiendo (9).

$$P_b(r) = \overline{P_r(p_{i,min})} - \max\{P_b(p_{i,R}), P_b(p_{i,L})\} \qquad (9)$$

Como excepción, si sólo se encuentra un único límite, el valor se guarda para mejor el modelado de la carretera pero sin calcular la puntuación del anillo $P_b(r)$. Esta excepción no puede ocurrir en dos anillos consecutivos ya que puede compromete el ángulo de búsqueda. El ángulo de búsqueda se actualiza usando los nuevos límites de la carretera.

D. Ajuste y modelado de la carretera

Una vez extraídos los límites izquierdos y derechos de cada nube de puntos, el siguiente paso es generar un modelo en 2D de la carretera para poder visualizar y supervisar los resultados, y más importante seguir el centro de la carretera. A pesar de ser B-Spline el modelo más empleado en el estado del arte para modelar la carretera ya que se obtiene como resultado unas curvas que ajustan los puntos de forma precisa y suave, este sistema no permite extrapolar datos. Es por ello que se ha escogido una regresión polinómica por mínimos cuadrados.

La regresión polinómica comienza con determinar el grado del polinomio. Analizando los puntos centrales en la carretera se obtiene el tipo de tramo: recta, curva o eslalon, es decir, como máximo se limita a cuatro el grado del polinomio, ya que en un tramo de 25 m no se suele encontrar un polinomio más complejo y el rendimiento recae notablemente si se sobreajusta la función. El siguiente paso es eliminar lo datos atípicos empleando la regresión en dos pasos. El primero paso es

Algorithm 2 Búsqueda de los límites

Input: $r_i = \{p_{i,1}, p_{i,2}, \ldots, p_{i,n_i}\}$
Parameters: s, n_i
Initialisation : $step = 0$
for $k = 1$ to n_i **do**
  $j = (i + step) \bmod n_i$
  if $P_r(p_{i,j})$ is a local minimum **then**
    save $P_r(p_{i,j})$ for road score
  end if
  if $P_r(p_{i,j}) < \varepsilon_b$ **then**
    replace $P_r(p_{i,j})$ with local minimum
    return j are referenced in the
  else
    if Clockwise detection **then**
      $step = step + 1$
    else
      $step = step - 1$
    end if
  end if
end for
return No boundary detected

aplicar la regresión eliminando aquellos puntos más alejados de la estimación, que superen un umbral ε_b:

$$|y_i - \widehat{y_i}| > \ \varepsilon_b \qquad (10)$$

Aplicando (10) en cada uno de los puntos, tanto a izquierda como derecha, se consigue una mayor precisión en la regresión. En la Fig. 4, en azul la regresión sin filtro y en verde aplicando el filtro de datos atípicos, se detalla la diferencia entre las dos regresiones.

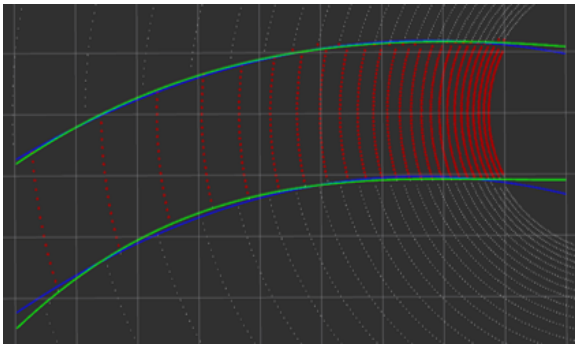


Fig. 4. Filtrado de puntos atípicos en entorno de simulación.

Finalmente, el ángulo central de la carretera se calcula para actualizar el ángulo de referencia θ_s que empleará la siguiente nube y como referencia para el control reactivo

III. Resultados

Los ensayos se han realizado en una pista de maniobras con un ancho de calzada relativamente uniforme de aproximadamente 7.5 m, con tramos de pendientes de hasta un 18% de inclinación, curvas abiertas, cerradas y/o peraltadas para

poder testear los escenarios posibles que un vehículo se puede encontrar.

El vehículo empleado, de dimensiones medias, 3.930 mm de longitud, 1,980 mm de anchura y 2,800 mm de altura, y una velocidad máxima de 20 km/h, se ha configurado para circular a una velocidad de entre 2 a 4 m/s, dependiendo del tramo de carretera. Los sensores de percepción 3D son:

- Dos LiDAR de 16 planos, modelo RS-LiDAR-16 con alcance de hasta 150 m y una precisión de 2 cm, generando 320,000 puntos/s. El LiDAR extremo izquierdo está orientado con un roll, pitch y yaw de 0.104, 1.166 y 1.016 rad respectivamente, mientras que el derecho está orientado a 0.084, 1.14 y -0.906 rad (ver Fig. 1).
- LiDAR de 64 planos, modelo OS-1 64 con alcance de hasta 120 m y una precisión media de 3 cm, generando 1,310,720 puntos/s. El LiDAR está orientado a -0.02, 0.12 y -0.002 rad (ver Fig. 1).

Los primeros ensayos (Fig. 5) en tramos rectos y de curva suave permite comprobar la efectividad del sistema para segmentar y modelar la carretera con un rango de 25 m, pudiendo hacer un seguimiento efectivo del centro de la calzada. Estos tramos se caracterizan por unos límites uniformes y con alteraciones suaves, cuyo elemento más desafiante es el nivel de resorte de los límites (de media 0,1 m). A pesar de esta dificultad, el sistema identifica correctamente los límites, descartando los falsos positivo (puntos superiores del límite izquierdo) mediante el filtro de puntos atípicos. Los sensores laterales permiten recoger la información más crítica en torno al vehículo, mientras que el central perfila el modelo hacia delante y hacia atrás, para estabilizar los puntos ciegos en la parte trasera-lateral del vehículo.

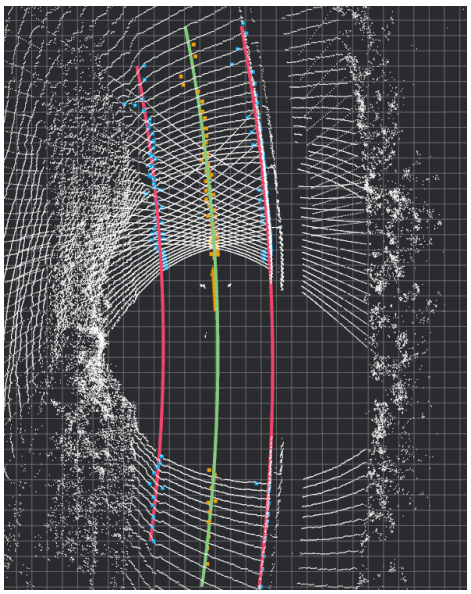


Fig. 5. Detección de la carretera con búsqueda de 360 grados.

Tramos más complejos como curvas con o sin peralte (Fig. 6) muestran la importancia del filtro de datos atípicos para corregir la selección de los límites cuando la diferencia el terreno arenoso con el asfalto dificulta notablemente la extracción de puntos. Sin embargo, gracias a la orientación de los LiDAR laterales se consiguen el número necesario de límites en los primeros 15m para generar una regresión válida. En esta imagen además podemos analizar la búsqueda de los anillos (línea azul) para estudiar la extracción de los límites.

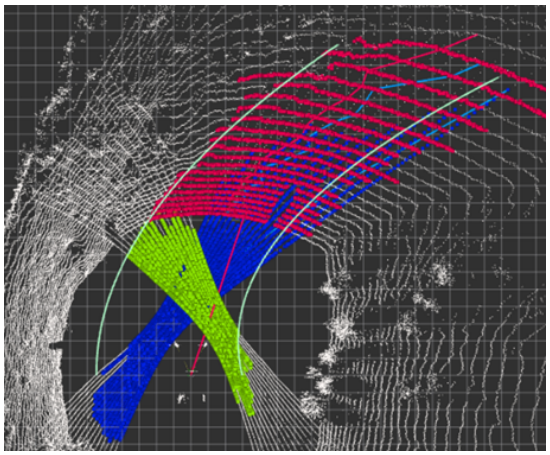


Fig. 6. Detección de la carretera en curva cerrada.

En la Fig. 7 se realiza una comparativa del seguimiento del centro de la carretera por la pista, en violeta el centro marcado manualmente y en rojo el recorrido actual del control reactivo alimentado por la detección de la carretera. Ambas marcas se superponen con una gran precisión, apreciándose una estabilidad significativa tanto en los tramos de rectas como en curvas abiertas, mientras que en los tramos de curvas cerradas el vehículo se abre (manteniendo un margen de seguridad) ya que el límite exterior se encuentra más cerca de los sensores y es más fácil de detectarlo.

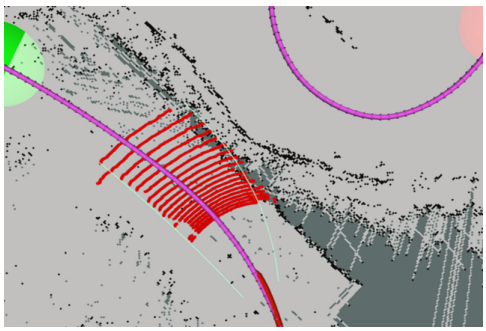


Fig. 7. Comparativa del seguimiento del centro de la carretera.

IV. Conclusión

El propósito de este trabajo es desarrollar una solución basada en percepción LiDAR para detectar la carretera en un entorno rural de bajo contraste y sin marcas viales como alternativa a los sistemas visión por computación. Por esta misma razón no se ha planteado un sistema híbrido que

combine los LiDAR y cámaras como se ha visto en el estado del arte.

Viendo los resultados obtenidos en las pistas de maniobras se puede confirmar que el algoritmo cumple de manera fiable los requisitos para la detección de la carretera y se establece como un sistema fiable que respalde la navegación autónoma del vehículo en caso de pérdida de señal GPS.

La segmentación de la carretera extrae información suficiente para abordar de forma estable tramos con con un límite borroso o poco pronunciado, curvas cerradas, secciones peraltadas y cambios de desnivel. Este sistema, debido a la búsqueda en un sólo canal de los límites, no es compatible con la bifurcaciones. Es necesario cambiar el enfoque de búsqueda y realizar la búsqueda por secciones para poder incorporar la búsqueda en varios canales y poder abordar las bifurcaciones correctamente.

Como líneas futuras de trabajo se plantea explotar el potencial del modelo probabilístico para fusionar los límites de todos los sensores dado su puntuación y tener en mayor cuenta aquellos límites con una menor probabilidad de pertenecer a la carretera. Adicionalmente, emplear un filtro condicional adaptativo al ángulo de búsqueda y al grado de confianza de la nube para optimizar el número puntos a analizar. Finalmente, incorporar el enfoque de búsqueda por secciones en un anillo para contemplar bifurcaciones y obstáculos en la carretera.

Agradecimientos

Investigación avalada por el gobierno de España a través de los proyectos de la CICYT (PID2019-104793RB-C31 y RTI2018-096036-B-C21) y de la Comunidad de Madrid a través de SEGVAUTO-4.0-CM (P2018/EMT-4362).

References

[1] D. Munoz, N. Vandapel, M. Hebert, *et al.*, "Directional associative markov network for 3-d point cloud classification," in *Fourth international symposium on 3D data processing, visualization and transmission*, pp. 65–72, Georgia Institute of Technology Atlanta, US, 2008.

[2] C. P. McElhinney, P. Kumar, C. Cahalane, and T. McCarthy, "Initial results from european road safety inspection (eursi) mobile mapping project," *Commission V Symposium*, vol. 38, no. 5, 2010.

[3] A. B. Hillel, R. Lerner, D. Levi, and G. Raz, "Recent progress in road and lane detection: a survey," *Machine vision and applications*, vol. 25, no. 3, pp. 727–745, 2014.

[4] S. Thrun, M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, G. Hoffmann, K. Lau, C. Oakley, M. Palatucci, V. Pratt, P. Stang, S. Strohband, C. Dupont, L.-E. Jendrossek, C. Koelen, and P. Mahoney, "Stanley: The robot that won the darpa grand challenge.," *J. Field Robotics*, vol. 23, pp. 661–692, 01 2006.

[5] C. Urmson, J. Anhalt, D. Bagnell, C. Baker, R. Bittner, M. Clark, J. Dolan, D. Duggins, T. Galatali, C. Geyer, *et al.*, "Autonomous driving in urban environments: Boss and the urban challenge," *Journal of Field Robotics*, vol. 25, no. 8, pp. 425–466, 2008.

[6] M. Quintana, J. Torres, and J. M. Menéndez, "A simplified computer vision system for road surface inspection and maintenance," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 3, pp. 608–619, 2015.

[7] S. Chatterjee, P. Saeedfar, S. Tofangchi, and L. M. Kolbe, "Intelligent road maintenance: a machine learning approach for surface defect detection.," in *ECIS*, p. 194, 2018.

[8] C. Yu and D. Zhang, "Road curbs detection based on laser radar," in *2006 8th international Conference on Signal Processing*, vol. 4, IEEE, 2006.

[9] N. Mattern, R. Schubert, and G. Wanielik, "High-accurate vehicle localization using digital maps and coherency images," in *2010 IEEE intelligent vehicles symposium*, pp. 462–469, IEEE, 2010.

[10] A. Barsi and C. Heipke, "Artificial neural networks for the detection of road junctions in aerial images," *International Archives of Photogrammetry Remote Sensing and Spatial Information Sciences*, vol. 34, no. 3/W8, pp. 113–118, 2003.

[11] S. Ferchichi and S. Wang, "Optimization of cluster coverage for road centre-line extraction in high resolution satellite images," in *IEEE International Conference on Image Processing 2005*, vol. 2, pp. II–201, IEEE, 2005.

[12] J. Siegemund, D. Pfeiffer, U. Franke, and W. Förstner, "Curb reconstruction using conditional random fields," in *2010 IEEE Intelligent Vehicles Symposium*, pp. 203–210, IEEE, 2010.

[13] L. Zhou and G. Vosselman, "Mapping curbstones in airborne and mobile laser scanning data," *International Journal of Applied Earth Observation and Geoinformation*, vol. 18, pp. 293–304, 2012.

[14] Y. Zhang, J. Wang, X. Wang, and J. M. Dolan, "Road-segmentation-based curb detection method for self-driving via a 3d-lidar sensor," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 12, pp. 3981–3991, 2018.

[15] P. Sun, X. Zhao, Z. Xu, R. Wang, and H. Min, "A 3d lidar data-based dedicated road boundary detection algorithm for autonomous vehicles," *IEEE Access*, vol. 7, pp. 29623–29638, 2019.

[16] B. Rodriguez-Cuenca, S. Garcia-Cortes, C. Ordóñez, and M. C. Alonso, "An approach to detect and delineate street curbs from mls 3d point cloud data," *Automation in Construction*, vol. 51, pp. 103–112, 2015.

[17] A. Jaakkola, J. Hyyppä, H. Hyyppä, and A. Kukko, "Retrieval algorithms for road surface modelling using laser-based mobile mapping," *Sensors*, vol. 8, no. 9, pp. 5238–5249, 2008.

[18] B. Yang, L. Fang, and J. Li, "Semi-automated extraction and delineation of 3d roads of street scene from mobile laser scanning point clouds," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 79, pp. 80–93, 2013.

[19] F. Xu, L. Chen, J. Lou, and M. Ren, "A real-time road detection method based on reorganized lidar data," *PloS one*, vol. 14, no. 4, 2019.

[20] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *International Journal of Robotics Research (IJRR)*, 2013.

[21] J.-Y. Sun, S.-W. Kim, S.-W. Lee, Y.-W. Kim, and S.-J. Ko, "Reverse and boundary attention network for road segmentation," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pp. 0–0, 2019.

[22] G. L. Oliveira, W. Burgard, and T. Brox, "Efficient deep models for monocular road segmentation," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4885–4891, IEEE, 2016.

[23] Z. Chen, J. Zhang, and D. Tao, "Progressive lidar adaptation for road detection," *IEEE/CAA Journal of Automatica Sinica*, vol. 6, no. 3, pp. 693–702, 2019.

[24] L. Caltagirone, M. Bellone, L. Svensson, and M. Wahde, "Lidar–camera fusion for road detection using fully convolutional neural networks," *Robotics and Autonomous Systems*, vol. 111, pp. 125–131, 2019.

[25] L. Xiao, B. Dai, D. Liu, T. Hu, and T. Wu, "Crf based road detection with multi-sensor fusion," in *2015 IEEE Intelligent Vehicles Symposium (IV)*, pp. 192–198, IEEE, 2015.

[26] K. A. Redmill, J. I. Martin, and U. Ozguiner, "Sensing and sensor fusion for the 2005 desert buckeyes darpa grand challenge offroad autonomous vehicle," in *2006 IEEE Intelligent Vehicles Symposium*, pp. 528–533, IEEE, 2006.

Modelado de la carga de vehículos eléctricos basada en agentes

Rafael Rus Rus
Departamento de Informática
Universidad Carlos III de Madrid
Leganés (Madrid), España
100363907@alumnos.uc3m.es

Agapito Ledezma Espino
Departamento de Informática
Universidad Carlos III de Madrid
Leganés (Madrid), España
ledezma@inf.uc3m.es

***Abstract* — En este trabajo se plantea la creación de un modelo de simulación de movilidad de vehículos privados, tanto eléctricos como clásicos, teniendo en cuenta múltiples variables de circulación y el comportamiento de los vehículos autónomos eléctricos a través del estudio del comportamiento de sus baterías y su relación con los centros de carga en la ciudad de Aranjuez (Madrid, España).**

Cabe destacar que la importancia de este trabajo reside en la creación de una herramienta de simulación capaz de simular y visualizar el tráfico de una ciudad y estudiar el comportamiento de los vehículos eléctricos autónomos en ella.

Keywords* — *Coche eléctrico, baterías, simulación, GAMA, QGIS

I. Introducción

Cada vez es más latente la necesidad de encontrar soluciones a los problemas más comunes relacionados con la conducción y la movilidad en nuestras ciudades a través de los vehículos privados (tráfico, accidentes, estrés, congestión, contaminación, …), y es por ello por lo que se está impulsando, cada vez más, el uso de las tecnologías referidas a los vehículos autónomos, concretamente a aquellos que funcionan mediante energía eléctrica.

No obstante, estas tecnologías aún no están lo suficientemente desarrolladas para obtener un nivel completo de autonomía de los vehículos que permita su uso de manera masiva en nuestras ciudades.

Hoy en día, se están llevando a cabo numerosos estudios y proyectos de diseño de vehículos autónomos, y una de las fases necesarias en la puesta a punto de este tipo de vehículos son las pruebas de funcionamiento en un simulador de tráfico.

Los simuladores de tráfico no sólo son una buena herramienta para desarrollar modelos de movilidad urbana dentro de una ciudad, sino que también permiten estudiar cómo se comportarían los vehículos circulando en estos modelos de movilidad.

Debido a esto último, en este trabajo se plantea la creación de un modelo de simulación para el estudio de la movilidad autónoma y eléctrica. Una simulación es el proceso de diseñar un modelo de un sistema real y llevar a cabo experimentos con él, con el fin de entender así su comportamiento y evaluar nuevas estrategias para el funcionamiento del sistema [1].

Así pues, se plantea el desarrollo de un simulador de tráfico que permita a los usuarios analizar el comportamiento de los vehículos dentro de una ciudad, así como la posibilidad de medir y manipular los parámetros más relevantes, integrando vehículos autónomos eléctricos y permitiendo el estudio de la optimización de carga de estos vehículos mediante el uso de baterías eléctricas y estaciones de recarga.

El simulador se realiza mediante un entorno de desarrollo de modelado y simulación de agentes denominado GAMA [2], y ha sido diseñado tomando como referencia la ciudad de Aranjuez (Madrid, España). Esta ciudad está actualmente inmersa en un proyecto de conversión en Smart City [3], en el que se incluye la instalación a lo largo de la ciudad de infraestructuras para la recarga de las baterías de estos vehículos eléctricos. No obstante, la propuesta es extrapolable a otras localizaciones.

II. Un modelo para el estudio de la movilidad autónoma y eléctrica

A. Herramientas de modelización

Las herramientas de diseño de frameworks [4] permiten crear modelos de simulación con agentes, permitiendo incluso la creación de agentes relacionados con el tráfico de vehículos y los nuevos métodos de transporte, lo cual nos puede ayudar para ver cómo se comportan los coches dentro de una ciudad o de un área urbana.

Consisten en entornos de desarrollo de modelo y simulación que combinan elementos de sistemas complejos, teoría de juegos, emergencia, sistemas multi-agente y programación evolutiva, en los cuales se simulan todas las operaciones simultáneas que realizan algunas entidades múltiples (agentes) en un intento de predecir las acciones de fenómenos complejos.

Se considera, pues, un proceso de predicción, debido a que los agentes de la simulación experimentarán a lo largo de la misma un proceso de aprendizaje, adaptación y reproducción constante, con el fin de predecir las siguientes acciones de los fenómenos que se están estudiando, y completar de esta forma sus propios intereses.

Dentro de estas herramientas, existen dos niveles de exposición: el micro (comportamiento de los individuos de la simulación) y el macro (patrones que surgen de la interacción de esos individuos).

De entre todas las herramientas de diseño de frameworks actuales, destacan: Repast [5], NetLogo [6], AirSim [7], Gazebo [8] y GAMA.

La plataforma de modelado y simulación utilizada en este proyecto para construir una simulación basada en agentes es GAMA, una plataforma, cuyo objetivo es proporcionar a los expertos de campo, modeladores e informáticos un entorno completo de desarrollo de modelado y diseño para construir simulaciones de múltiples agentes espacialmente explícitas.

XXX-X-XXXX-XXXX-X/XX/$XX.00 ©20XX IEEE

Entre sus características principales, podemos destacar las siguientes [9]:

- Puede ser usado para múltiples dominios de aplicación, entre ellos el sector del transporte automovilístico (sector en el cual se centra este proyecto).
- Tiene su propio lenguaje de programación (llamado GAML). Consiste en un lenguaje intuitivo y de alto nivel basado en agentes, que se ejecuta en una máquina virtual de Java similar al programa Eclipse y es considerado de código abierto.
- Permite crear instancias de agentes a partir de cualquier conjunto de datos, incluidos datos GIS (sistemas de información geográfica) [10] y cuenta con una poderosa interfaz de usuario declarativa.

B. Proceso de creación del modelo

El proyecto desarrollado se puede estructurar en dos partes bien diferenciadas, la generación del mapa de la ciudad para su integración en la simulación y el desarrollo del modelo de la simulación.

Generación del mapa de la ciudad: para ello se utiliza una herramienta de mapeado como por ejemplo QGIS [1] construyéndose poco a poco el mapa de la ciudad utilizando capas para distinguir los agentes que se utilizarán luego en el simulador de GAMA.



Fig. 1. Muestra del mapa de un grafo direccional realizado mediante QGIS

Se ha creado una capa correspondiente a las vías de circulación del mapa (un grafo direccional por el que se definen parámetros como el sentido de circulación, el número de carriles, la velocidad máxima permitida, etc…) y otra relacionada con las intersecciones que se vayan a utilizar para unir esas calles (nodos que pueden definirse como intersecciones, pasos de cebra, semáforos, ceda el paso, etc…).

El proyecto desarrollado está pensado para que se puedan incluir todas estas características en el modelo del simulador de la plataforma GAMA utilizando toda esa información en la simulación.

Otro aspecto que se ha considerado en el mapa de la ciudad es la localización de todas las estaciones de recarga para vehículos autónomos eléctricos.

Su incorporación permite que, en el momento de la simulación, en el cual la carga de los vehículos eléctricos haya superado el umbral que se ha utilizado para establecer la condición de batería baja, dichos vehículos se moverán hacia la estación más cercana de acuerdo con sus cualidades y necesidades (tiempo necesario para la recarga, tipo de conector, capacidad, etc.).

	Carga normal		Carga semirrápida		Carga rápida
Tensión máx.	230 V monofásico		400 V trifásico		500 V continua
Corriente máx.	16 A	32 A	32 A	63 A	125 A
Potencia máx.	3,7 kW	7,4 kW	21 kW	43 kW	50 kW
Conector	Tipo 1, tipo 2, schuko	Tipo 1 y tipo 2	Tipo 2	Tipo 2	CHAdeMO, Combo
Ubicación	Viviendas, garajes privados...		Centros comerciales, vía pública...		Puntos estratégicos en carretera
Horas*	8 h	4 h 30 min	1 h 30 min	45 min	30 min

Fig. 2. Tipos de conectores y estaciones de recarga

Para llevar a cabo este trabajo, ha sido necesario realizar un estudio previo sobre cómo es el funcionamiento de los vehículos autónomos eléctricos y concretamente de sus baterías.

Para medir la carga de una batería (lo que comúnmente se conoce como "porcentaje de una batería") se ha utilizado el cálculo de variación del estado de carga (al cual, a partir de ahora, se referirá como SoC) [11].

Los fabricantes recomiendan que los valores óptimos de carga para una batería son entre el 20%-80% de su carga [12], por lo que, dentro del modelo desarrollado, se partirá de la idea en la que todos los vehículos autónomos eléctricos tendrán un valor de carga siempre entre esos valores.

A su vez, se ha decidido considerar el límite de aviso de batería baja en el 30%, de manera que cuando el SoC de un vehículo de la simulación llega a dicho umbral en el proceso de descarga de su batería, éste cambia su ruta hacia la estación más cercana.

También se ha añadido una particularidad y es que, cuando el SoC del vehículo tiene unos valores demasiado bajos (caso hipotético en el que el vehículo se mueve hacia la estación, todavía no ha llegado y su SoC es inferior al 10%) su probabilidad de ruptura aumenta con cada ciclo de simulación.

Para calcular la variación del estado se ha utilizado el método de *Coulomb Counting* (recuento de Coulomb) que permite medir la corriente de descarga de una batería y de integrar la corriente de descarga a lo largo del tiempo para estimar el SoC.

Existen varios métodos de estimación del SoC de las baterías, pero de entre todos ellos se ha decidido utilizar el método de *Coulomb Counting* debido a que su fórmula era la más apropiada para las características extraídas del modelo de batería utilizada. De todas formas, se podría emplear cualquier otro método. Este se calcula mediante la ecuación de *Coulomb Counting* [13], la cual se define a continuación (1):

$$SoC(t) = SoC(t-1) + \frac{i(t)}{C} d \qquad (1)$$

- C es la capacidad de la batería

[1] Versión 3.10.11

- *i(t)* es la intensidad eléctrica suministrada por la batería en amperios con respecto al tiempo de simulación

Basándose en la ecuación (1), es posible determinar la variación del estado de carga de la batería mediante esta otra fórmula del incremento de la variación del estado de carga (2):

$$\Delta SoC = \frac{\int i(t)dt}{C_{nom}} \quad (2)$$

En cuanto al modelo de batería utilizado, se ha decidido usar el del vehículo Hyundai IONIQ [14].

Tipo de batería	Capacidad nominal	Eficiencia	Tensión	Capacidad cargador
Batería de iones de litio	28 kWh	11,5 kWh / 100 km	360 V	6,6 kWh

Fig. 3. Características del modelo IONIQ

Para el cálculo del SoC se usa la siguiente lógica: cuando el vehículo se encuentre en movimiento y su batería se esté descargando, se tiene que cumplir que (3):

$$Soc_{final} = SoC_{inicial} - \Delta SoC \quad (3)$$

Mientras tanto, cuando el vehículo se encuentre conectado a un poste o estación de recarga (4):

$$Soc_{final} = SoC_{inicial} + \Delta SoC \quad (4)$$

Esta variación del estado de carga se mide en cada ciclo de tiempo en la simulación y depende de una variable llamada "step" (duración de paso o ciclo de simulación). Por defecto, esta variable, "step", es igual a un segundo, pero se puede configurar dentro de la simulación, dependiendo del experimento que se quiera probar.

Si se quiere que la simulación vaya más lenta, habría que disminuir el valor de la variable "step"; mientras que para aumentar su velocidad habría que agrandar su valor.

Por ejemplo, si la variable "step" es igual a 0,7 segundos (reducción de la velocidad con respecto a la normal de 1 seg.) y teniendo en cuenta las propiedades del modelo de batería IONIQ, se tendría que la variación del estado de carga cuando se descarga una batería se calcula (5):

$$SoC_{final} = SoC_{inicial} - \frac{\int_0^{0,7} \frac{Eficiencia(t)}{Tensión} dt}{C_{nom}}$$

$$= SoC_{inicial} - \frac{\int_0^{0,7} \frac{\frac{11,5\ kWh}{100\ km}(t)}{360\ V} dt}{C_{nom}} \quad (5)$$

Y la variación del estado de carga cuando se está cargando una batería se calcula (6):

$$SoC_{final} = SoC_{inicial} + \frac{\int_0^{0,7} \frac{C_{cargador}(t)}{Tensión} dt}{C_{nom}}$$

$$= SoC_{inicial} + \frac{\int_0^{0,7} \frac{6,6\ kWh\ (t)}{360\ V} dt}{C_{nom}} \quad (6)$$

Desarrollo del modelo de la simulación: para la realización de éste se ha hecho uso de la plataforma GAMA (versión 1.8.1.) y se ha subdividido a su vez en dos partes: la edición del modelo y la ejecución de experimentos.

El modelo está planteado para poder ser editado por un usuario experto o modelador encargado de utilizar la simulación para sus propios usos.

El modelo está dividido en los siguientes agentes computacionales: especie global/"world" (el agente principal que se ocupa de lanzar el experimento y de montar el mapa generado en QGIS, el cual debe ser importado siempre en formato ESRI Shapefile (.shp)) y las microespecies (el resto de los agentes de la simulación: intersecciones/"crossroads", carreteras/"roads", vehículos autónomos normales/"normal Cars" y vehículos autónomos eléctricos/"electricCars").

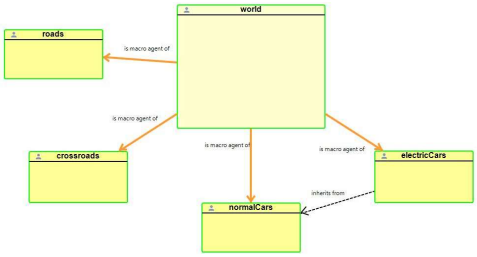


Fig. 4. Diagrama de la interacción entre los agentes

Como se puede ver en la anterior figura, los atributos de todas las microespecies son heredados de la especie global. A su vez, la microespecie "electricCars" hereda los atributos de la microespecie "normalCars, ya que ambos son vehículos y presentan muchas propiedades en común.

El funcionamiento consiste en que los vehículos autónomos se mueven por el grafo, formado por las carreteras y las intersecciones, desde un punto aleatorio del mapa a otro; generándose, por cada vehículo, una ruta específica.

Para la programación de dicha ruta se ha hecho uso de una librería de GAMA denominada "habilidad de conducción avanzada" [15], la cual se ocupa de diseñar la ruta eligiendo el mejor camino posible (camino óptimo) del grafo.

La "habilidad de conducción avanzada" es lo suficientemente versátil como para ser utilizada con la mayoría de los datos GIS de carreteras clásicos, en particular, los datos OSM (los dos mapas obtenidos en formato ESRI Shapefile).

Se usa un formato clásico para las carreteras y los nodos, en el que cada camino es una polilínea compuesta por secciones de camino (segmentos) y está formado por un nodo de destino y un nodo de origen; y cada nodo conoce todos sus caminos de entrada y salida.

El funcionamiento general de esta habilidad es pensar que los caminos se consideran como dirigidos y los vehículos se mueven a través de ellos mediante una acción denominada *do drive*.

A su vez, una carretera puede estar compuesta por varios carriles y los vehículos pueden cambiar de carril en cualquier momento. Otra propiedad de la vía que se tiene en cuenta es la velocidad máxima autorizada en la misma.

Cada agente conductor tiene una trayectoria planificada (denominada *current_path*) que consta de una sucesión de aristas desde su punto de partida hasta su punto de llegada en el mapa (ambos aleatorios).

Para la construcción de dicha ruta se han establecido dos cualidades para la toma de decisiones de optar por un camino u otro: las velocidades máximas permitidas en cada calle/carretera; y, en caso de llegar a un nodo con dos o más bifurcaciones, primero se elige el carril de acuerdo con la

densidad del tráfico, teniendo en cuenta que hay un sesgo para el carril más a la derecha.

Así pues, para cada vehículo se va a diseñar una ruta desde un punto aleatorio a otro de la ciudad, la cual va a seguir en cada ciclo de la simulación, y durante su movimiento el vehículo tomará todas las decisiones de tráfico que sean adecuadas (velocidad, semáforos, preferencias etc..). Además, el vehículo se adapta a las velocidades de cada carretera de la ruta, teniendo en cuenta sus propiedades: aceleración máxima, longitud, etc. A partir de ellas se calcula la velocidad esperada del agente conductor, que es la velocidad en tiempo simulado a la que se mueve el agente entre dos nodos del grafo.

Cuando un vehículo llega a su destino final, éste se elimina y automáticamente se regenera otro con otras propiedades y que tiene también que realizar una ruta aleatoria.

En cuanto a los vehículos autónomos eléctricos, éstos se van a comportar de manera idéntica, con la particularidad de que, en el momento en el que se encuentren con batería baja (SoC inferior al 30%), cambian su ruta para dirigirse hacia la estación de recarga más cercana compatible con su tipo de cargador.

Una vez hayan llegado a la estación, se detienen en ella, cargan sus baterías durante un tiempo determinado (o hasta que su SoC se encuentre al 80% de su carga, que como ya se ha visto, es el valor máximo más óptimo) y, a continuación, reanudan sus rutas hacia los objetivos que tenían anteriormente.

Con respecto a la ejecución de experimentos, al poner en marcha uno de ellos, el modelo redirige al usuario a la vista "perspectiva de simulación" que permite ejecutar/pausar un experimento, ejecutarlo ciclo por ciclo, reiniciarlo, visualizar y modificar las características de los agentes en cualquier momento de la simulación.

La simulación tiene un reloj interno que mide tanto el tiempo real como el tiempo simulado en el experimento [16].

A su vez, se ha introducido una variable temporal que depende de la variable "step" y que asigna de forma aleatoria una hora concreta del día dentro de la simulación. De esta forma, es posible definir unos horarios de tráfico para, por ejemplo, modificar los tiempos de espera de los vehículos en los semáforos y los pasos de cebra durante las horas del día con mayor tráfico de peatones. De esta forma, se simulan de manera más realista los tiempos de espera de los vehículos.

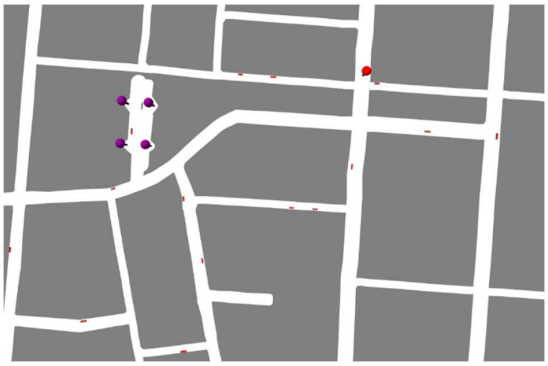


Fig. 5. Muestra de un trozo del mapa de un experimento del modelo desarrollado

III. Resultados del modelo

Para demostrar el correcto funcionamiento del modelo, se han llevado a cabo una serie de pruebas (16 experimentos concretamente) en dos computadores distintos (un ordenador portátil Lenovo ThinkPad L460 y un HP Pavilion 25-ec1005ns), todos ellos con la siguiente dinámica:

1) Evaluación de tiempos (reales y simulados) en cada uno de los experimentos, e interpretación de los resultados de carga en velocidad y memoria dependiendo del computador y del número de agentes.

El objetivo es comparar los tiempos de cómputo de los dos ordenadores tras realizar 2.000 ciclos y quedarse con el mejor para las siguientes partes de la evaluación. Tras dicho análisis, se ha determinado que un ordenador con mejor procesador va más rápido en tiempo real de la simulación.

También hay que tener en cuenta que, cuantos más agentes haya en el experimento simulado, más tarda en tiempo real y simulado con respecto a otras simulaciones. Así mismo, al modificar la variable "step" (duración de paso), el tiempo simulado también varía.

2) Análisis del movimiento y de las propiedades de un vehículo aleatorio de la simulación dentro de un periodo entre dos ciclos. Desde la perspectiva de simulación se va a medir cómo es el comportamiento de un vehículo autónomo aleatorio (normal o eléctrico) entre los ciclos 100 y 200 de la simulación. Tras dicho análisis, se ha demostrado que se cumplen los objetivos que se propusieron para el proyecto.

3) Verificación del correcto comportamiento de los agentes "normalCars" y "electricCars" (recorrido, respecto de las normas de circulación establecidas, etc.…); y medición de los tiempos de espera en señales de parada como semáforos, cedas el paso y stops.

4) Evaluación de las gráficas de la variación del estado de carga de las baterías de los vehículos autónomos eléctricos de la simulación e interpretación de los datos. Comprobando que los valores del SoC siempre van a estar entre los rangos de valores óptimos (20% - 80%), así como:

a) El hecho de que a medida que el vehículo descarga su batería, el SoC disminuye de forma gradual.

b) El hecho de que cuando se visualice una línea constante en la gráfica significa que el vehículo está detenido (semáforo, stop, etc…).

c) El hecho de que cuando el SoC de un vehículo ha superado el 30%, éste cambia la ruta hacia la estación y se pone a cargar su batería.

d) El hecho de que cuando un vehículo haya llegado a su destino, se produce una subida o una bajada considerable y se genera un nuevo vehículo en otro punto del mapa con una nueva ruta y un nuevo valor de SoC (mayor o menor que el anterior).

e) El hecho de que cuando un vehículo supere el 10% de su SoC, su probabilidad de ruptura aumenta mucho y eso hace que se pueda romper.

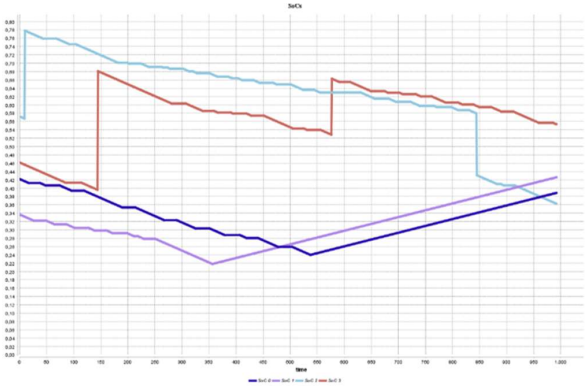


Fig. 6. Ejemplo de gráfica de la variación del estado de carga

A modo de resumen, tras haber hecho los análisis de los experimentos, se pueden citar las siguientes deducciones:

En relación con los tiempos y las velocidades de procesamiento al ejecutar una simulación del modelo diseñado, se ha visto que cuanto mayor sean las dimensiones del mapa que se vaya a utilizar y cuantos más agentes móviles (vehículos autónomos eléctricos y no) se utilicen en un experimento, mayor es el tiempo real en segundos.

Como ya se ha visto antes, el tiempo simulado está completamente relacionado con el valor de la variable "step" (paso de ciclo), la cual se ha ido modificando en las diferentes pruebas para ver cómo se alteran las variables y las propiedades de los agentes cuando se ejecuta un experimento.

A raíz de todo ello, se llega a la conclusión de que para ejecutar un experimento del modelo desarrollado es recomendable utilizar unos valores de la variable "step" adecuados, entre 0,9 y 1,5 aproximadamente, ya que sino la simulación o funciona muy despacio o muy deprisa.

Además, este valor se puede ajustar dependiendo del número de agentes que vaya a haber en el experimento, lo cual viene a decir que se puede utilizar cualquier mapa de cualquier tipo en formato ESRI Shapefile.

En cuanto al resto de habilidades y funcionalidades del modelo desarrollado, quizás la que se tiene que remarcar más tras la realización de los experimentos es la de los tiempos de espera en señales de parada. En numerosas ocasiones se ha visto que se han generado algunos atascos cuando se intentan crear simulaciones con muchos vehículos, concretamente en puntos con señales de pasos de cebra y semáforos.

Dicho esto, se recomienda reducir los tiempos de espera del modelo en dichas señales si lo que se quiere es mejorar el tráfico de la ciudad y así evitar la formación de estos atascos.

De la misma forma, cuando se ejecutan experimentos con una enorme cantidad de vehículos autónomos eléctricos, se suelen generar algunos atascos en las estaciones de recarga que hay por la ciudad. Esto se debe a que quizás no hay suficientes estaciones de recarga dentro de la ciudad madrileña de Aranjuez y, por ello, deberían instalarse más como medida para así poder evitar este problema.

IV. Conclusiones

La principal conclusión que se puede obtener es que se han cumplido en este proyecto todos los objetivos marcados:

- Se ha creado un modelo de simulación de movilidad de vehículos privados, tanto eléctricos como clásicos, los cuales se comportan teniendo en cuenta una serie de múltiples variables de circulación (respeto de todas las señales de parada como semáforos, cedas el paso, ..., de las velocidades máximas permitidas, de la distancia de seguridad con respecto a otros vehículos, …).
- Se ha diseñado un modelo de simulación basado en agentes de tráfico extrapolable a cualquier ciudad, el cual puede ser editado por cualquier usuario experto o modelador con el fin de llevar a cabo sus propias pruebas y simulaciones.
- Se ha estudiado el comportamiento de los vehículos autónomos eléctricos a través del estudio del comportamiento de sus baterías y su relación con los centros de carga en la ciudad de Aranjuez (Madrid, España).
- Se ha empleado de forma exitosa una herramienta de diseño de frameworks basada en agentes, lo cual ayuda mucho no solo en este campo del automovilismo, sino también de cara a poder desarrollar simulaciones de diversos ámbitos. La facilidad del modelado de GAMA (con su lenguaje GAML) da lugar a ello.

Por último, a la hora de mencionar algunos de los posibles trabajos futuros, se podrían citar aquellos destinados de cara a hacer el modelo mucho más realista.

Para ello, se sugieren algunas de las siguientes ideas:

1) Añadir al modelo líneas de transporte público como autobuses o taxis, los cuales ya van a tener rutas predefinidas.

2) Incorporar la visualización de peatones. Consistiría en crear otro agente que represente a los peatones, y que éstos se muevan por calzadas y crucen semáforos y pasos de cebra.

3) Generar los edificios del área urbana del mapa de la simulación. Representarlos en 3D para producir un mejor impacto visual dentro de la simulación. La mejor manera de implementar esto último es añadiendo la componente Z en el diseño del mapa de la ciudad, posibilitando además el hecho de que se puedan introducir rampas de subida y bajada que afecten al nivel de carga de la batería y a sus condiciones físicas.

4) Considerar la posibilidad de generar agentes para poder permitir ver los niveles de contaminación de la ciudad. Dado que se está trabajando con un entorno de modelado y diseño de frameworks que permite crear simulaciones de todo tipo y juntarlas, se podría intentar combinar este modelo con otro que permita detectar los niveles de contaminación del área estudiada y mostrarlo mediante nubes de diferente color para referirse a cada uno de los contaminantes estudiados.

5) Estudiar la posibilidad de incluir agentes conductores que se muevan con autonomía. Como se ha podido observar, todos los vehículos dentro de la simulación siguen la misma estrategia cuando, en un entorno real, problablemente cada vehículo seguiría una estrategia diferente. Así pues, se podría configurar para elegir una estrategia de conducción distinta para cada vehículo.

6) El modelo de la batería es muy lineal, ya que se trata de una primera aproximación (utilizando el método de *Coulomb Counting*) que presenta muchas limitaciones. Por ello, se podría mejorar el modelo para futuros usos dentro del simulador incluyendo otras baterías (como las híbridas) u otros métodos de estimación de la variación del estado de carga.

7) Cabe añadir también que se podrían incluir otras variables aleatorias para hacer que el comportamiento de la batería sea más realista, como sus condiciones físicas.

Bibliografía

[1] Paragon Homepage, https://www.paragon.com.br/es/academico-2/que-es-simulacion/, last accessed 2021/05/21.

[2] GAMA Homepage, https://gama-platform.github.io/, last accessed 2021/02/04.

[3] Aranjuez se convertirá en una 'Smart City' con la ayuda de la URJC, Universidad Rey Juan Carlos, 2018, https://www.urjc.es/todas-las-noticias-de-actualidad/2945-aranjuez-se-convertira-en-una-smart-city-con-la-ayuda-de-la-urjc, last accessed 2021/02/04.

[4] F. Sancho Caparrini, Simulación Social: Una introducción, 2019, http://www.cs.us.es/~fsancho/?e=52, last accessed 2021/02/05.

[5] Repast Homepage, https://repast.github.io/, last accessed 2021/02/04.

[6] NetLogo Homepage, https://ccl.northwestern.edu/netlogo/, last acceseed 2021/02/04.

[7] AirSim Homepage, https://microsoft.github.io/AirSim/, last accessed 2021/02/04.

[8] Gazebo Homepage, http://gazebosim.org/tutorials?tut=model_structure, last accessed 2021/02/04.

[9] GAMA Homepage - Introduction, https://gama-platform.github.io/wiki/Home, last accessed 2021/02/16.

[10] Wikipedia Homepage - Geographic information system, https://en.wikipedia.org/wiki/Geographic_information_system, last accessed 2021/02/05.

[11] J. Insa, Como entender las tensiones de batería y estado de carga, Monsolar, https://www.monsolar.com/blog/como-entender-las-tensiones-de-bateria-y-estado-de-carga/, last accessed 2021/02/05.

[12] E. D. Kostopoulos, G. C. Spyropoulos, and J. K. Kaldellis, Real-world study for the optimal charging of electric vehicles, Energy Reports, vol. 6, pp. 418–426, 2020.

[13] W.-Y. Chang, The State of Charge Estimating Methods for Battery: A Review, ISRN Appl. Math., pp. 1–7, Jul. 2013.

[14] Diariomotor Homepage - Hyundai IONIQ, https://www.diariomotor.com/coche/hyundai-ioniq/, last accessed 2021/02/05.

[15] GAMA Homepage - Advanced Driving Skill, https://gama-platform.github.io/wiki/UsingDrivingSkill, last accessed 2021/02/25.

[16] GAMA Homepage - Controls of experiments, https://gama-platform.github.io/wiki/MenusAndCommands, last accessed 2021/03/02.

Sistema de detección de fatiga basado en deep learning y lógica difusa

Elena Magán López
Departamento de Informática
Universidad Carlos III de Madrid
Madrid, España
emagan@inf.uc3m.es

María Paz Sesmero Lorente
Departamento de Informática
Universidad Carlos III de Madrid
Madrid, España
msesmero@inf.uc3m.es

Araceli Sanchis de Miguel
Departamento de Informática
Universidad Carlos III de Madrid
Madrid, España
masm@inf.uc3m.es

***Abstract*—En este trabajo se propone un sistema que puede integrarse en un sistema avanzado de asistencia a la conducción (ADAS) de un vehículo para detectar la fatiga en sus etapas tempranas y alertar al conductor de la situación, previniendo así posibles accidentes. Este sistema utiliza técnicas de *deep learning* para detectar las características faciales y el estado del conductor, y utiliza estos parámetros como entrada para un sistema de inferencia difusa. Éste utiliza lógica difusa para calcular el grado de cansancio del conductor, que el ADAS podrá utilizar para lanzar alarmas si se cumplen determinadas condiciones. Aunque se necesita continuar trabajando en el sistema para mejorar su eficacia en la detección de fatiga en personas con características diversas, comienza de forma prometedora al evitar en su mayoría los falsos positivos y reconocer algunos estados de cansancio.**

***Palabras clave*—lógica difusa, *deep learning* (aprendizaje profundo), redes de neuronas convolucionales, visión por computador, detección de fatiga, cansancio, seguridad en la conducción**

I. Introducción

El cansancio, definido como la necesidad de dormir, puede provocar síntomas que afectan gravemente al rendimiento de una persona durante la ejecución de una o varias tareas; por ejemplo, disminución de concentración y reflejos, fragmentos de tiempo en los que no percibe lo que le rodea, o microsueños (parpadeos con una duración superior a 500ms). [1]. Es más, la falta de sueño continuada puede provocar pérdidas de capacidad de razonamiento similares a las que se producen en un estado de embriaguez [2], [3].

En el entorno de la conducción, estos síntomas son extremadamente peligrosos, ya que aumentan las probabilidades de que un conductor se salte señales de tráfico, se salga de su carril o incluso se choque con su vehículo y provoque un accidente de tráfico [4].

La premisa de este trabajo es la siguiente: mediante una cámara colocada en el vehículo, se obtendrán imágenes del conductor tomadas de frente, que serán analizadas mediante técnicas de inteligencia artificial como el *deep learning* y la lógica difusa para detectar si éste está cansado o no, de tal forma que se le pueda alertar para prevenir accidentes.

El trabajo presentado en este artículo ha sido posible gracias a los proyectos de investigación TRA2016-78886-C3-1-R, PID2019-104793RB-C31, RTI2018-096036-B-C22, PEAVAUTO-CM-UC3M subvencionados por el Gobierno de España y al proyecto EPUC3M17 subvencionado por la comunidad de Madrid.

Los algoritmos de *deep learning* se caracterizan por utilizar redes de neuronas cuyos modelos están formados por grandes cantidades de capas [5], y permiten realizar el proceso de extracción de características de los datos de forma automática [6]. Dentro de los algoritmos de *deep learning* encontramos las redes de neuronas convolucionales (CNNs), que obtienen muy buenos resultados dentro del campo de la visión por computador al ser capaces de encontrar patrones y reconocer características en imágenes [7].

En este trabajo se utilizan técnicas de *deep learning* y visión por computador para obtener características a partir de la imagen de un conductor, como que el conductor esté bostezando o parpadee demasiado lento. Los parámetros obtenidos se introducirán en un sistema de inferencia de lógica difusa, donde se procesarán para valorar el cansancio del conductor.

La lógica difusa utiliza reglas del tipo "SI p ENTONCES q", pero, mientras que en la lógica clásica p y q sólo pueden ser verdadero o falso, en lógica difusa una cláusula puede ser parcialmente cierta [8]. Esto nos permite tratar con información que no es del todo precisa, lo cual es muy útil a la hora de definir las características que se detectan en el conductor (por ejemplo, al valorar si un cierto número de parpadeos implica que el conductor esté parpadeando "rápido" o "despacio").

Además, puesto que el consecuente también puede ser verdadero o falso en cierto grado, esto nos permitirá calcular el grado de cansancio estimado del conductor en lugar de determinar únicamente si está cansado o no, lo que puede ayudar a evitar falsos positivos. De esta forma, en el momento en el que se estime que el conductor está demasiado cansado para conducir, se le podría avisar para que detuviese el vehículo, protegiendo así su seguridad y la de su entorno. Así este sistema se podrá integrar en el futuro en un ADAS para monitorizar al conductor, considerando los valores de su cansancio para generar las correspondientes alarmas.

Este trabajo se organiza de la siguiente forma: la sección II presenta trabajos recientes relacionados con la detección de fatiga, así como ejemplos de diferentes técnicas aplicadas a la seguridad en la conducción. El diseño del sistema de detección de fatiga se describe en la sección III, mientras que la sección IV recoge los resultados obtenidos. Finalmente, la sección V presenta las conclusiones y propone posibles líneas de investigación futuras.

II. Estado del arte

A la hora de detectar el cansancio de un conductor, y en-función de la naturaleza de los datos usados en la bibliografía, se recogen dos enfoques diferentes: por una parte, tenemos sistemas que observan el estado del vehículo para valorar el estado de fatiga del conductor, mientras que, por otra, nos encontramos con sistemas que utilizan parámetros medidos sobre el propio conductor.

Entre los trabajos dedicados al estudio del estado del vehículo y su relación con el cansancio, las medidas más comunes que se analizan son las variaciones del volante o la capacidad del conductor de mantenerse en su carril [9], [10], [11]. En [12] se utilizan otros parámetros del vehículo como la posición del vehículo o el giro del volante, y combina varias medidas para conseguir un sistema más fiable. Sin embargo, aunque la disminución en el rendimiento del conductor sobre tareas que requieran de su habilidad puede ser un síntoma de fatiga, éste aparece de forma tardía y no puede utilizarse para detectar las fases tempranas del cansancio [13].

Es por ello que en este trabajo nos centraremos en detectar los primeros síntomas del cansancio a partir del estado del propio conductor. Una de las formas más precisas de estimar la fatiga del conductor se obtiene mediante la utilización de electroencefalogramas (EEG) en combinación con electrooculogramas (EOG) [14], pero en escenarios de conducción reales este tipo de sistemas suele ser rechazado por los conductores. Su principal desventaja es que requieren de la colocación de electrodos alrededor de los ojos y en la cabeza del conductor, lo que deriva en sistemas muy intrusivos que generan incomodidad y rechazo. Debido a esta limitación, los sistemas más utilizados son aquellos en los que el estado del conductor se percibe a través de las imágenes captadas mediante una cámara colocada en el vehículo.

Entre los trabajos que siguen este enfoque, existe mucha variedad respecto a los parámetros estudiados y a las técnicas utilizadas para obtenerlos. En [15], por ejemplo, en primer lugar se obtienen los *landmarks* de la cara del conductor, esto es, una serie de puntos que permiten localizar los elementos más importantes de la cara: ojos, nariz, boca, cejas, y contorno facial. Mediante estos *landmarks* se calculan varios parámetros, como el porcentaje de cierre del ojo (PERCLOS), y luego se introducen estos parámetros en una máquina de soporte vectorial (SVM) para clasificar si el conductor está cansado o no. En [16], en cambio, se utilizan mapas de profundidad y *deep learning* para la detección de fatiga. En concreto, utiliza dos CNNs: una CNN espacial, que detecta la posición de los objetos, y una CNN temporal, que busca información entre dos fotogramas consecutivos. Mediante estas dos CNNs es capaz de calcular los vectores de movimiento de un fotograma a otro, lo que le permite detectar bostezos incluso cuando el conductor se pone la mano delante de la boca.

La lógica difusa [8] se convierte en una herramienta muy útil a la hora de crear sistemas que ayuden a proteger al conductor. Por una parte, esto se debe a la facilidad con la que se pueden crear reglas precisas con resultados fácilmente entendibles, y por otra, a la reducida carga computacional que requieren para funcionar, que permite que el sistema pueda funcionar en tiempo real. Además, los ejemplos no se reducen a la detección de fatiga; en [17], por ejemplo, se propone un sistema de alertas basado en lógica difusa que genera alarmas para avisar al conductor de situaciones peligrosas.

Dentro de los trabajos que combinan lógica difusa y detección de fatiga, en [18] encontramos un sistema que propone analizar la boca y los ojos del conductor, midiendo cómo de abiertos están para poder juzgar si el conductor está cansado o no, y si detecta cansancio durante varios segundos, lanza una alarma. En [19], en cambio, se utilizan medidas que tienen en cuenta no sólo el estado del conductor en ese instante, sino también el comportamiento que ha tenido durante el último minuto. Así, calcula parámetros como el PERCLOS, la frecuencia de parpadeo o la posición de la cara, todo ello durante una ventana de 60 segundos. Después, introduce estos parámetros en un sistema de inferencia difusa que contiene 32 reglas diferentes, y con ello calcula el nivel de cansancio del conductor.

Si bien ambos trabajos informan de buenos resultados, es importante destacar que los experimentos fueron realizados sobre personas que fingían estar cansadas. Al simular estas situaciones, los sujetos de estudio tienden a exagerar los gestos y a mostrar síntomas claramente visibles, lo que provoca que el sistema desarrollado no sea tan fiable en entornos reales.

Para evitar el problema de los datos simulados, en este trabajo se utilizará la base de datos *UTA Real-Life Drowsiness Dataset* (UTA-RLDD) [20]. Esta base de datos contiene vídeos de 60 personas realizando una tarea simple (leer o ver algo en su ordenador), grabados de frente y con una duración de 10 minutos por grabación. Estos vídeos están clasificados en función del estado en el que se encontraban los sujetos cuando se grabaron (alerta, poco atento, o cansado), y cada persona tiene un vídeo para cada categoría. En este aspecto, para realizar comparaciones resulta especialmente interesante el trabajo de [21], ya que se implementa un sistema de reconocimiento de fatiga mediante una CNN y se prueba sobre la base de datos de UTA-RLDD.

III. Descripción del sistema

En esta sección se describe el sistema diseñado para la detección de fatiga de un conductor.

Como se ha mencionado previamente, el sistema utiliza imágenes frontales del conductor captadas a través de una cámara, que podría estar, por ejemplo, en el salpicadero del vehículo. Esta imagen es preprocesada y tratada mediante técnicas de *deep learning*, y con ello se obtienen una serie de características numéricas que se podrán introducir en el sistema de inferencia difusa (FIS). Tras ello, el FIS devolverá un valor numérico que representará el cansancio que se ha estimado que tiene el conductor, y finalmente este valor nos permitirá lanzar una alerta.

La Fig. 1 muestra un esquema que resume el proceso que sigue el sistema. A continuación, se explican las 4 etapas en las que se divide el sistema.

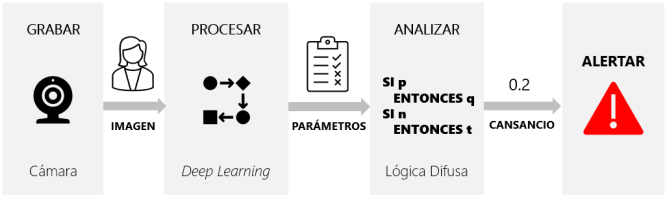


Fig. 1: Etapas del sistema.

A. *Captación de las imágenes*

A la hora de obtener las imágenes del conductor, es relevante determinar el número de fotogramas por segundo (FPS) que transmite la cámara: una tasa de FPS demasiado alta impedirá que el sistema funcione en tiempo real debido a la gran cantidad de datos que tendrá que analizar por segundo, pero una tasa de FPS demasiado baja podría afectar negativamente al rendimiento del sistema.

Es necesario que haya suficientes fotogramas por segundo como para detectar todos los detalles de la secuencia de imágenes, donde cobran especial importancia los parpadeos debido a la brevedad con la que éstos ocurren. Puesto que la duración media de un parpadeo oscila entre los 100-400ms [22], para este trabajo se ha optado por utilizar una tasa de 10 FPS, que será suficiente para detectar los parpadeos y permitirá no sobrecargar demasiado el sistema.

B. *Preprocesamiento de las imágenes*

Los parámetros que nos interesa conocer a partir de la imagen del conductor son los siguientes:

- Número de parpadeos
- Duración media de parpadeos
- Número de microsueños (parpadeos con duración superior a 500ms)
- Número de bostezos
- Tiempo que se pasa el conductor bostezando
- Tiempo que se pasa el conductor hablando

El primer paso para obtener estos datos será localizar la cara del conductor. Para esto se utiliza la librería DLIB [23], y en concreto su detector facial, que obtiene las coordenadas en las que se encuentra la cara del conductor gracias la utilización de Histogramas de Gradiente Orientado (HOG) [24]. Con esta información, y utilizando la implementación de DLIB para la detección de *landmarks*, podemos obtener también los *landmarks* de la cara del conductor, y con ello la posición de sus ojos. El detector de *landmarks* de DLIB representa los rasgos de una persona mediante 68 puntos, de los cuales 12 de ellos representan los ojos (6 puntos por ojo). La Fig. 2 muestra un ejemplo de detección de *landmarks*.

Para calcular los parámetros relacionados con los ojos y los parpadeos será necesario calcular, en primer lugar, cómo de abierto o de cerrado se encuentra el ojo. Para ello seguimos la misma técnica que utilizan en [25]: obtener el *Eye Aspect Ratio* (EAR), calculando la distancia entre el párpado de arriba y el párpado de abajo y dividiéndola por la anchura del ojo, obteniendo valores de abertura que suelen oscilar en el rango

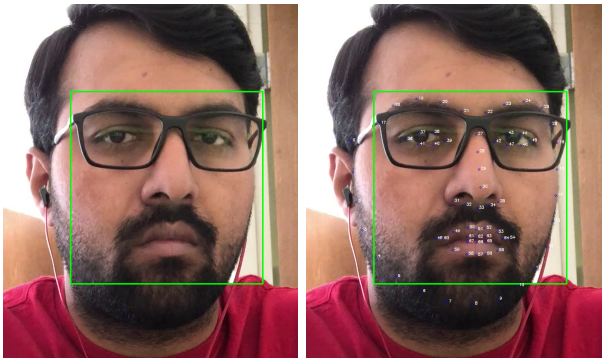


Fig. 2: (izq.) Cara detectada por el algoritmo de detección facial de DLIB. (dcha.) *Landmarks* detectados por el algoritmo de detección de *landmarks*.

de (0.16, 0.36). De forma experimental se ha determinado un umbral de 0.22, de tal modo que siempre que el porcentaje de abertura sea menor a 0.22 se considerará que el conductor tiene los ojos cerrados. Calculando cuántas veces y durante cuánto tiempo se producen los parpadeos es como obtenemos las tres primeras medidas.

A continuación, queremos utilizar la cara del conductor para detectar si está bostezando, está hablando, o no está haciendo ninguna de las dos cosas. Para ello extraemos la parte de la imagen con la cara, la redimensionamos a 64x64px y realizamos una ecualización del histograma de la imagen para realzar los contrastes de la imagen y eliminar detalles innecesarios. Además, aplicamos la resta de la media RGB de *ImageNet*, una técnica que disminuye la probabilidad de que la iluminación de una imagen afecte a la clasificación, para así poder aplicar *transfer learning* (transferencia de aprendizaje) y partir de modelos ya preentrenados para la clasficación de imágenes. La Fig. 3 muestra un ejemplo del proceso.

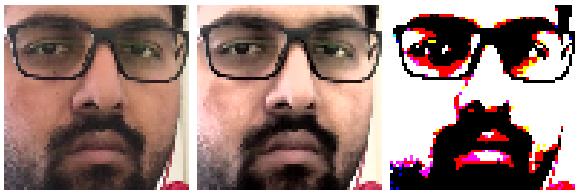


Fig. 3: (izq.) Cara original. (centro) Con equalización del histograma. (dcha.) Con equalización del histograma y resta de la media RGB.

Esta imagen se le pasa a una CNN que hemos entrenado específicamente para reconocer si una persona está bostezando, está hablando o conduce con normalidad. Dicha red parte del modelo *EfficientNetB0* [26], reutilizando su arquitectura y utilizando los pesos entrenados para la clasificación de objetos de la base de datos *ImageNet*, y le añade capas propias para entrenarlo en el reconocimiento de las acciones que nos interesan. Para el entrenamiento de esta red de neuronas se ha utilizado la base de datos *YawDD* [27], que dispone de vídeos de 30 personas realizando estas tres acciones al volante.

Para el entrenamiento de esta CNN se utilizaron los vídeos grabados desde una cámara situada en el salpicadero, enfrente del conductor. Los vídeos de 18 de esas personas se utilizaron para la fase de entrenamiento, 6 se usaron para la validación y 6 como conjunto de test.

Gracias a este clasificador podemos calcular cuánto tiempo se ha pasado el conductor bostezando o hablando durante el último minuto, obteniendo así todos los parámetros que necesita el sistema de inferencia difusa.

C. Sistema de inferencia difusa

Para la detección del cansancio se ha diseñado un sistema de inferencia difusa de tipo Mamdani [28], para el cual tendremos que definir: entradas, salidas, y reglas.

1) Entradas: Cada entrada del sistema debe representarse mediante una variable, que a su vez deberá tener asignados uno o más *fuzzy sets* que representen los posibles valores que puede tomar (por ejemplo, la variable "número de parpadeos", que recibiría un número entero, tendría los *fuzzy sets* "bajo", "normal" y "alto"). Los *fuzzy sets*, a su vez, vienen definidos por una función de pertenencia $\mu\text{A}(x)$, que asigna a una entrada con valor x un valor de pertenencia al *fuzzy set* A. Dicho valor de pertenencia será un número entre 0 y 1, donde 0 significa que es completamente falso que el valor x pertenezca al *fuzzy set* A y 1 que es completamente cierto.

Para este trabajo se han utilizado funciones de pertenencia triangulares (tri) y trapezoidales (trap). En ambos casos las funciones están definidas por una serie de puntos que se conectan linealmente, pero mientras que en la función triangular hay tres puntos (con valores de pertenencia 0, 1 y 0) en la función trapezoidal hay 4 puntos (con valores de pertenencia 0, 1, 1 y 0). La Tabla I muestra las variables utilizadas junto a sus respectivos *fuzzy sets*.

TABLA I: Entradas FIS

Nombre de la variable	**Fuzzy sets**	
	Nombre	***Función de pertenencia***
No. de parpadeos (blinks)	bajo	$tri(0, 0, 10)$
	normal	$tri(0, 10, 20)$
	alto	$trap(10, 20, 50, 50)$
Duración media de un parpadeo (avg_blink)	bajo	$trap(0, 0, 0.10, 0.30)$
	normal	$trap(0.10, 0.30, 0.40, 0.60)$
	alto	$trap(0.40, 0.60, 60, 60)$
Número de microsueños (microsleeps)	bajo	$tri(0, 0, 2)$
	alto	$trap(1, 3, 60)$
Número de bostezos (yawns)	alto	$trap(0, 2, 30, 30)$
Tiempo bostezando (avg_yawn)	alto	$trap(0, 10, 60, 60)$
Tiempo hablando (avg_talk)	alto	$trap(0, 20, 60, 60)$

2) Salidas: Al igual que en las entradas, las salidas de un FIS vienen determinadas por variables definidas mediante *fuzzy sets*. El resultado se calculará como la pertenencia a los diferentes *fuzzy sets*, y despues será *defuzzificado* para obtener un único valor numérico que represente a esa variable (en nuestro caso, un número en el intervalo [0,1] indicando el cansancio del conductor). La variable de salida se define como se indica en la Tabla II.

TABLA II: Salidas FIS

Nombre de la variable	**Fuzzy sets**	
	Nombre	***Función de pertenencia***
Cansancio	bajo	$tri(0, 0, 0.5)$
	alto	$tri(0.5, 1, 1)$

El método utilizado para la *defuzzificación* es el Centro de Gravedad (COG), un método que, en función de las activaciones de los *fuzzy sets* de salida, calcula el centro de gravedad del área bajo las funciones de pertenencia [29].

3) Reglas: Finalmente, las reglas relacionan las variables de entrada con las variables de salida del sistema. En nuestro caso, debemos definir las condiciones que reflejan el estado de cansancio de un conductor. De esta forma, se definen 7 reglas diferentes:

1) **SI** (avg_blink ES alto) **ENTONCES** (cansancio ES alto)
2) **SI** (microsleeps ES alto) O (blinks ES alto) **ENTONCES** (cansancio ES alto)
3) **SI** (avg_blink ES bajo) Y (blinks ES alto) **ENTONCES** (cansancio ES alto)
4) **SI** (yawns ES alto) O (avg_yawn ES alto) **ENTONCES** (cansancio ES alto)
5) **SI** (avg_blink ES normal) O (blinks ES normal) **ENTONCES** (cansancio ES bajo)
6) **SI** (avg_blink ES bajo) Y (blinks ES bajo) Y (microsleeps ES bajo) **ENTONCES** (cansancio ES bajo)
7) **SI** (avg_talk ES alto) **ENTONCES** (cansancio ES bajo)

D. Alertas

En base a los resultados obtenidos por el sistema de lógica difusa, se han determinado las siguientes condiciones para hacer saltar una alerta:

- El conductor se detecta como "cansado" cuando la salida del sistema de inferencia difusa es superior a 0.65
- Si se detecta como "cansado" al conductor durante al menos 30 de los últimos 60 segundos, salta una alarma
- Si estas dos condiciones se siguen cumpliendo, no se vuelve a lanzar la alarma
- Cuando dejan de cumplirse las condiciones para que salte la alarma, y después vuelve a detectarse el cansancio, sí podrá saltar otra alarma

IV. Resultados

Las pruebas se han realizado sobre una subparte de los datos proporcionados por la base de datos UTA-RLDD: en concreto, se ha probado el sistema sobre 25 vídeos de 12 sujetos diferentes. En 12 de esos vídeos (denominados vídeos *awake*) los sujetos afirmaban estar despiertos, mientras que en los otros 13 (que se denominarán vídeos *drowsy*) afirmaban sentirse cansados.

En la Fig. 4 se pueden observar las predicciones de cansancio para cada uno de los vídeos, representando cada vídeo con una línea de diferente color. En el gráfico azul se muestran

las predicciones de cansancio calculadas sobre los vídeos *awake*, mientras que en el naranja se observan esas mismas predicciones pero calculadas sobre los vídeos *drowsy*. Como se puede observar, en los vídeos *drowsy* hay muchos picos y valores altos, lo que es un buen indicativo de la precisión del sistema. Sin embargo, aunque en los vídeos *awake* los valores de cansancio tienden a ser más bajos, también podemos observar que aparecen algunos picos que podrían provocar falsos positivos. Para evitar esto se configuran los valores del umbral y del tiempo de detección, tal y como se han explicado en la sección III-D.

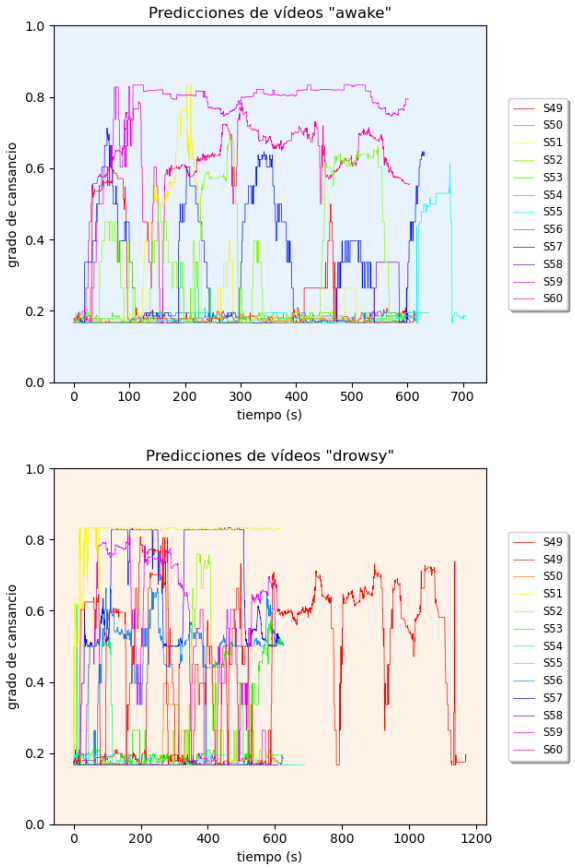


Fig. 4: (arriba) Resultados al predecir el cansancio en los *awake videos*. (abajo) Resultados al predecir el cansancio en los *drowsy videos*.

Las variaciones del umbral de alarma, así como del tiempo mínimo que debe pasar detectando cansancio para que se active, son elementos que afectan directamente a los resultados obtenidos: a menor tiempo de detección y mayor umbral, más probable es que salten alarmas. Esto contribuye a advertir al conductor de forma más temprana y en más situaciones de cansancio, pero también provoca que aparezcan más falsos positivos cuando no está cansado. Por ejemplo, disminuir el umbral de 0.65 a 0.6 y bajar el tiempo de detección a 25 segundos provoca que se salte una alarma correctamente en un vídeo *drowsy* en el que antes no saltaba, pero también provoca que se active en dos vídeos *awake* en los que no le corresponde.

Es prioritario evitar los falsos positivos, ya que, si el conductor se ve molestado por un sistema que le avisa cuando no debe, es altamente probable que directamente desactive el sistema. Es por ello que se ha optado por elegir las combinaciones de umbral y tiempo de detección que más precisión tienen y, entre ellas, se ha seleccionado la que menos falsos positivos tiene. Teniendo esto en cuenta, la Tabla III muestra el número de alarmas activadas durante las pruebas.

TABLA III: Alarmas lanzadas

UTA-RLDD Número de sujeto	**Awake videos**		**Drowsy videos**	
	Alarmas lanzadas	***Marca de tiempo 1ª alarma (s)***	***Alarmas lanzadas***	***Marca de tiempo 1ª alarma (s)***
49	-	-	1 3	219.5 252.4
50	-	-	-	-
51	-	-	1	53.1
52	-	-	1	394
53	-	-	-	-
54	-	-	-	-
55	-	-	-	-
56	-	-	-	-
57	-	-	2	139.3
58	-	-	-	-
59	1	103.7	3	97.7
60	3	114.8	-	-

Como se puede observar, mientras que en los vídeos *awake* la precisión es del 83%, y sólo se producen falsos positivos en dos sujetos, en los vídeos *drowsy* el rendimiento baja y sólo detecta correctamente el cansancio en el 46% de los casos, obteniendo así una precisión global del 64%.

Es necesario continuar con la investigación para tratar de maximizar los casos de cansancio que se detectan, estudiando las diferencias entre las demostraciones de cansancio que pueden presentar unas personas y otras. Si observamos el caso del sujeto 49, podemos intuir que la detección del cansancio dependerá de la persona que se esté estudiando, ya que se disponía de dos vídeos diferentes de este sujeto para el estado *drowsy* y en ambos se ha detectado con éxito el cansancio.

También resulta llamativo el caso del sujeto 60, que, mientras que en estado *awake* activa una alarma tres veces, en el estado *drowsy* no llega a activar ninguna. Observando detenidamente las características de estos vídeos, se detecta que es el único sujeto con rasgos asiáticos sobre el que hemos realizado las pruebas. En su vídeo *awake*, observamos que detecta los ojos de forma precisa, pero al calcular el EAR se interpreta que tiene los ojos cerrados cuando realmente no lo están, indicando que es necesario considerar las diferencias en los rasgos faciales entre razas. En su vídeo *drowsy*, en cambio, la iluminación es distinta y no se detectan correctamente los *landmarks* de los ojos, por lo que realmente no llegan a detectarse los parpadeos, lo que explica ese doble fallo.

A la hora de comparar estos resultados con trabajos previos, en el trabajo [21] afirman lograr una precisión del 91.8% mediante una CNN entrenada específicamente sobre los vídeos

de este dominio. Es interesante mencionar que en este trabajo no se realizan las pruebas sobre los vídeos completos, sino que sólo se extrae y se clasifica un fotograma al azar de cada vídeo, con lo que no podemos realizar una comparación sobre el rendimiento del sistema a lo largo del tiempo. Por otra parte, en ese mismo trabajo se compara con una implementación de [30] ejecutada sobre el dataset UTA-RLDD, que logra una precisión del 63%, similar a los resultados obtenidos aquí.

V. Conclusiones y trabajos futuros

En este trabajo se ha realizado una primera aproximación a un sistema de detección de fatiga mediante la utilización de técnicas de *deep learning* y lógica difusa. Este sistema es capaz de advertir al conductor de un estado de cansancio en algunas situaciones, y consigue evitar lanzar alertas innecesarias en la mayoría de ocasiones. Es necesario continuar para mejorar la detección de las situaciones en las que el conductor está cansado y el sistema actual no le advierte.

El sistema propuesto ha sido validado sobre un dominio en el que los sujetos han clasificado su estado de cansancio mediante su propia percepción. Esta forma de evaluar es más compleja pero también más realista que la basada en vídeos en los que los protagonistas simulan la fatiga.

Para mejorar el sistema será necesario estudiar a fondo qué tipos de síntomas de cansancio pueden presentar las personas, y especialmente la diferencia de síntomas que puede haber entre unas y otras. Con esto, se podrán pulir y mejorar las reglas del sistema de inferencia difuso (añadiendo más variables de entrada y más reglas, modificando las funciones de pertenencia...), y con ello adaptar también los umbrales a partir de los cuales se activan las alarmas.

Además, una vez mejorado el sistema de detección de fatiga basado en el estado del conductor, sería útil complementarlo mediante la utilización de medidas provenientes del vehículo, como por ejemplo, el número de veces que el conductor se ve obligado a corregir su trayectoria para mantenerse en el carril.

Referencias

[1] DAVID F. DINGES. An overview of sleepiness and accidents. *Journal of Sleep Research*, 4(s2):4–14, 1995.

[2] Drew Dawson and Kathryn Reid. Fatigue, alcohol and performance impairment. *Nature*, 388:235, 08 1997.

[3] Ann M Williamson, Anne-Marie Feyer, Richard P Mattick, Rena Friswell, and Samantha Finlay-Brown. Developing measures of fatigue using an alcohol comparison to validate the effects of fatigue on performance. *Accident Analysis & Prevention*, 33(3):313–326, 2001.

[4] Sónia Soares, Tiago Monteiro, António Lobo, António Couto, Liliana Cunha, and Sara Ferreira. Analyzing driver drowsiness: From causes to effects. *Sustainability*, 12(5), 2020.

[5] Samira Pouyanfar, Saad Sadiq, Yilin Yan, Haiman Tian, Yudong Tao, Maria Presa Reyes, Mei-Ling Shyu, Shu-Ching Chen, and S. S. Iyengar. A survey on deep learning: Algorithms, techniques, and applications. *ACM Comput. Surv.*, 51(5), September 2018.

[6] Maryam Najafabadi, Flavio Villanustre, Taghi Khoshgoftaar, Naeem Seliya, Randall Wald, and Edin Muharemagic. Deep learning applications and challenges in big data analytics. *Journal of Big Data*, 2, 12 2015.

[7] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. *Commun. ACM*, 60(6):84–90, May 2017.

[8] L.A. Zadeh. Fuzzy logic. *Computer*, 21(4):83–93, 1988.

[9] Jarek Krajewski, David Sommer, Udo Trutschel, David Edwards, and Martin Golz. Steering wheel behavior based estimation of fatigue. In *Proceedings of the Fifth International Driving Symposium on Human Factors in Driver Assessment, Training and Vehicle Design*, pages 118–124, 10 2017.

[10] Fabian Friedrichs and Bin Yang. Drowsiness monitoring by steering and lane data based features under real driving conditions. In *2010 18th European Signal Processing Conference*, pages 209–213, 2010.

[11] Anthony D. McDonald, Chris Schwarz, John D. Lee, and Timothy L. Brown. Real-time detection of drowsiness related lane departures using steering wheel angle. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 56(1):2201–2205, 2012.

[12] Sajjad Samiee, Shahram Azadi, Reza Kazemi, Ali Nahvi, and Arno Eichberger. Data fusion to develop a driver drowsiness detection system with robustness to signal loss. *Sensors*, 14(9):17832–17847, 2014.

[13] Ji Hyun Yang, Zhi-Hong Mao, Louis Tijerina, Tom Pilutti, Joseph F. Coughlin, and Eric Feron. Detection of driver fatigue caused by sleep deprivation. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 39(4):694–705, 2009.

[14] D. Sommer and M. Golz. Evaluation of perclos based current fatigue monitoring technologies. In *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*, pages 4456–4459, 2010.

[15] Yuan Gao and Changyuan Wang. Fatigue state detection from multi-feature of eyes. In *2017 4th International Conference on Systems and Informatics (ICSAI)*, pages 177–181, 2017.

[16] Xiaoxi Ma, Lap-Pui Chau, and Kim-Hui Yap. Depth video-based two-stream convolutional neural networks for driver fatigue detection. In *2017 International Conference on Orange Technologies (ICOT)*, pages 155–158, 2017.

[17] Elena Magán, Agapito Ledezma, Paz Sesmero, and Araceli Sanchis. Fuzzy alarm system based on human-centered approach. In *Proceedings of the 6th International Conference on Vehicle Technology and Intelligent Transport Systems - VEHITS,*, pages 448–455. INSTICC, SciTePress, 2020.

[18] Tayyaba Azim, M. Arfan Jaffar, and Anwar M. Mirza. Fully automated real time fatigue detection of drivers through fuzzy expert systems. *Applied Soft Computing*, 18:25–38, 2014.

[19] Luis Bergasa, Jesús Nuevo, Miguel-Angel Sotelo, Rafael Barea, and María Guillén. Real-time system for monitoring driver vigilance. *IEEE Transactions on Intelligent Transportation Systems*, 7:63–77, 01 2006.

[20] Reza Ghoddoosian, Marnim Galib, and Vassilis Athitsos. A realistic dataset and baseline temporal model for early drowsiness detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.

[21] Reza Tamanani, Radu Muresan, and Arafat Al-Dweik. Estimation of driver vigilance status using real-time facial expression and deep learning. *IEEE Sensors Letters*, 5(5):1–4, 2021.

[22] Harvey Richard Schiffman. *Sensation and perception: An integrated approach, 3rd ed*. 1990.

[23] Davis E. King. Dlib-ml: A machine learning toolkit. *Journal of Machine Learning Research*, 10:1755–1758, 2009.

[24] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 886–893 vol. 1, 2005.

[25] T. Soukupová and Jan Cech. Eye-blink detection using facial landmarks. In *Proceedings of the 21st Computer Vision Winter Workshop*, pages 22–29, 2016.

[26] Mingxing Tan and Quoc V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. *CoRR*, abs/1905.11946, 2019.

[27] Shabnam Abtahi; Mona Omidyeganeh; Shervin Shirmohammadi; Behnoosh Hariri. Yawdd: Yawning detection dataset, 2020.

[28] E.H. Mamdani and S. Assilian. An experiment in linguistic synthesis with a fuzzy logic controller. *International Journal of Man-Machine Studies*, 7(1):1–13, 1975.

[29] Werner Van Leekwijck and Etienne E. Kerre. Defuzzification: criteria and classification. *Fuzzy Sets and Systems*, 108(2):159–178, 1999.

[30] Bhargava Reddy, Ye-Hoon Kim, Sojung Yun, Chanwon Seo, and Junik Jang. Real-time driver drowsiness detection for embedded system using model compression of deep neural networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 438–445, 2017.

Sistemas empotrados para el reconocimiento cognitivo de conductores

Roberto Móstoles Rodríguez
Applied Artificial Intelligence Group (GIAA)
Universidad Carlos III de Madrid
Madrid, España
roberto.mostoles@alumnos.uc3m.es

Adrián Losada Casado
Applied Artificial Intelligence Group (GIAA)
Universidad Carlos III de Madrid
Madrid, España
adrian.losada@alumnos.uc3m.es

Juan Pedro Llerena Caña
Applied Artificial Intelligence Group (GIAA)
Universidad Carlos III de Madrid
Madrid, España
jllerena@inf.uc3m.es

Jesús García Herrero
Applied Artificial Intelligence Group (GIAA)
Universidad Carlos III de Madrid
Madrid, España
jgherrer@inf.uc3m.es

José Manuel Molina López
Applied Artificial Intelligence Group (GIAA)
Universidad Carlos III de Madrid
Madrid, España
molina@ia.uc3m.es

***Resumen*—El estado anímico de un conductor influye en la toma de decisiones y por consiguiente es una variable para tener en cuenta en los sistemas de prevención activa de accidentes. La identificación de emociones es ampliamente estudiada desde la psicología y aplicada a la conducción, entre otros campos, desde la percepción del individuo. Gran porcentaje de accidentes son derivados de estados anímicos concretos, por lo que disponer de sistemas automáticos de identificación de emociones resultan esenciales para nuevos sistemas de prevención activa de accidentes. En este trabajo se propone la detección facial y la clasificación de emociones asociadas desde los paradigmas de la visión por computador de alto nivel utilizando dos sistemas empotrados de bajo coste y tres métodos de detección y clasificación diferentes realizando un análisis desacoplado de las fases de detección y clasificación. Los resultados obtenidos demuestran que la combinación de técnicas clásicas de procesamiento de imágenes y arquitecturas de aprendizaje profundo pueden mejorar el rendimiento global del sistema integrador en las tareas de detección facial y reconocimiento de emociones a través de pistas faciales.**

***Palabras clave*—vehículos inteligentes, reconocimiento facial, identificación de emociones, visión por computador, aprendizaje automático**

I. Introducción

Desde organismos estatales como la Dirección General de Tráfico (DGT) se trabaja activamente para implementar técnicas y procedimientos de seguridad vial. Hay que tener en cuenta que un automóvil es un sistema que engloba el vehículo, la carretera y al conductor, siendo el factor humano asociado al conductor uno de los elementos menos predecibles y más frágil del sistema. Uno de los factores ampliamente estudiados asociados a los conductores son las causas de accidentes [1] entre los que podemos destacar la fatiga o el estrés los cuales incrementan hasta un 28% el riesgo de accidentes tal y como se muestra en [2]. Hay que tener en cuenta que cualquier situación relacionada con el tráfico genera emociones, existiendo multitud de bibliografía dedicada a la psicología aplicada a la conducción [3]–[5]. Esta bibliografía, generalmente, se centra en la identificación de las emociones por parte del propio conductor con el objetivo de poder controlarlas consiguiendo así un comportamiento racional que derive en una conducción más segura.

Algunas de las técnicas de seguridad vial con más importancia durante los últimos años son los sistemas ADAS (*Advanced Driver Assistance Systems*), encargados de proporcionar asistencia a la conducción mediante el aumento de la seguridad activa de los vehículos. De acuerdo con la DGT, el uso de estos sistemas en España podría reducir el riesgo de siniestro en un 57% [6]. Su funcionamiento se basa en la recopilación de información del entorno, bien mediante cámaras u otro tipo de sensores, proporcionando así información al conductor con la que poder actuar en consecuencia. Ejemplos de estos sistemas son la cámara de marcha atrás, los sistemas de ayuda en pendiente o la detección de fatiga, siendo estos algunos de los factores humanos que pueden afectar a la tarea de la conducción [7].

En Shi [8], se implementa un sistema inteligente de percepción de la fatiga y la ira de conductores basado en la clasificación de señales de la actividad electrodérmica (EDA) de la piel utilizando una máquina de vectores de soporte (SVM). Para la experimentación utilizan un conjunto de individuos expuestos a un entorno de simulación, obteniendo una precisión en clasificación de un 85%. En [9] se utilizan modelos ocultos de Markov (HMM) para tratar de evitar robos de vehículos mediante la detección del comportamiento y de la cara del conductor. Si este no se corresponde con el piloto habitual y su comportamiento tampoco se basa en el del modelo previsto, se procede con el apagado completo del vehículo, evitando así el robo del mismo. Por último, en [10] se trata de detectar la posibilidad de sufrir un accidente grave o leve mediante la monitorización del entorno de conducción y el análisis de las expresiones faciales del piloto. Para ello, sincronizando ambos niveles de datos, mediante el uso de la técnica estadística chi-cuadrado, se extraen las expresiones faciales más comunes las cuales, al clasificarse mediante distintas técnicas de aprendizaje automático, ofrecen como predicción las posibilidades de sufrir un accidente de mayor o menor grado.

En este trabajo se propone el desarrollo e implementación de un sistema basado en técnicas de aprendizaje automático para tareas de análisis cognitivo llevadas a cabo de forma eficiente. El sistema base tiene como finalidad analizar el estado emocional de un conductor humano en tiempo real a partir de las pistas expresivas de su rostro, haciendo hincapié en la utilización de técnicas computacionalmente eficientes en su implementación.

El sistema final se implementa en dos equipos empotrados de tipo Raspberry Pi 4B y NVIDIA Jetson Nano mediante los que se compara el rendimiento de la propuesta en dos de las arquitecturas de mayor alcance comercial en la actualidad, una de ellas basada en código abierto, siendo la segunda de tipo privado. Las variables de interés en cada caso son la eficiencia computacional y la precisión en detección, para lo que se

XXX-X-XXXX-XXXX-X/XX/$XX.00 ©20XX IEEE

ofrece una evaluación completa de algunas de las técnicas de detección de mayor relevancia para el alcance del estudio.

El presente trabajo está organizado del siguiente modo: la sección II desarrolla la propuesta del estudio que se lleva a cabo. En la sección III se presenta la experimentación desarrollada con la que comprobar el rendimiento de la propuesta. La sección IV analiza los resultados obtenidos durante la experimentación. Por último, la sección V, se encarga de exponer las conclusiones obtenidas, así como posibles trabajos futuros.

II. Propuesta de estudio

La capacidad de realizar inferencia emocional sobre la expresividad facial del conductor en tiempo real es, potencialmente, una de las mejores ayudas a la conducción que se pueden integrar en vehículos actuales: la reactividad emocional del conductor está directamente influenciada por los eventos que ocurren en su entorno, de forma que resulta indispensable disponer de medios que permitan capturar y analizar la expresividad emocional a medida que se produce. Para legitimar la utilización de estos sistemas en el mundo real, y más concretamente en el entorno de la ayuda a la conducción, los subsistemas encargados tanto de la detección facial como del reconocimiento emocional asociadas deben ser eficaces y eficientes en el cómputo, de otra forma, la información generada será inservible en el mejor de los casos y perjudicial para el conductor en el peor de ellos. Para este fin, el presente estudio ofrece una comparación del rendimiento de distintas estrategias de detección facial y reconocimiento emocional en sistemas *hardware* (HW) compactos.

A. Recursos hardware

El foco del estudio propuesto recae en la utilización de sistemas compactos en la labor de evaluación del estado anímico del conductor. La reducida huella de estos sistemas facilita su implementación en la mayoría de los escenarios, lo que junto a su consumo reducido permiten una integración con mínimo impacto negativo en el entorno. Para este fin, en el estudio se ha evaluado el rendimiento de dos sistemas compactos Raspberry Pi 4B [11] y NVIDIA Jetson Nano [12]. El sistema Raspberry incorpora una CPU ARM64 Arm-Cortex A72, 4GB de memoria RAM y 64GB de almacenamiento. El sistema NVIDIA cuenta con una CPU ARM64 Arm-Cortex A57, 2GB de memoria RAM y 64GB de almacenamiento, además de una GPU NVIDIA Maxwell. Ambos sistemas cuentan con versiones compatibles de Linux Debian.

B. Entorno de la propuesta

Desde la perspectiva de las estrategias inteligentes para la identificación de objetos en imágenes, actualmente existen dos vertientes principales: Modelos de aprendizaje profundo basados en la utilización de arquitecturas neuronales para identificar jerárquicamente las características más significativas, y modelos de aprendizaje automático clásicos que emplean técnicas de procesamiento numérico tradicionales para realizar la inferencia de características sobre la imagen. En vista de esta tendencia, se propone una evaluación de las cualidades de ambas tipologías en el ámbito de la identificación eficiente de rostros empleando alguna de las implementaciones más relevantes en cada ámbito.

En la actualidad, las técnicas de aprendizaje profundo de mayor relevancia científica son aquellas basadas en arquitecturas convolucionales, con las que se consiguen destacar características de interés para la identificación de objetos en una imagen procesando recursivamente sus componentes mediante operaciones de convolución de su tensor de píxeles equivalente. Para la evaluación de este conjunto de técnicas se ha partido de modelos pre-entrenados de las arquitecturas de redes Single Shot MultiBox Detector (SSD) [13], Region-based Fully Convolutional Networks (RFCN) [14] y Faster R-CNN [15]. Los modelos de referencia se han extraído del repositorio oficial de Tensorflow, Detection Model Zoo [16].

Las técnicas tradicionales para la detección de rostros en imágenes se centran en la identificación de patrones relativos a la diferencia en intensidad en distintas regiones de una imagen. Bajo este paradigma, la inferencia se realiza a partir de la combinación y procesamiento de características de bajo nivel sobre el color de la imagen. De esta forma, para la evaluación de las técnicas tradicionales se ha propuesto la utilización de modelos basados en el concepto de Histogramas de Gradientes Orientados (HOG), y de características Haar procesadas con el algoritmo Viola-Jones [17] para la identificación de objetos en una imagen. En el caso de los modelos basados en HOG se emplean vistas alternativas de la imagen que la presentan como un gradiente de intensidades, lo que ayuda a identificar contornos y accidentes de interés sobre el objeto a detectar [18]. Por otra parte, los modelos basados en el algoritmo de Viola-Jones (VJ) obtienen la representación de estos contornos mediante la aplicación directa de filtros sobre la imagen, permitiendo una evaluación rápida de características localizadas a cambio de una penalización en la robustez ante variaciones en las características del objeto a detectar.

Los modelos presentados resuelven el problema de la identificación de un rostro en una imagen cualquiera. Sin embargo, la inferencia automática del estado anímico de un conductor requiere de la integración de una arquitectura adicional encargada de la labor de reconocimiento emocional a partir del rostro detectado. Para tal fin, se propone el uso de una red convolucional basada en la arquitectura VGG-16 [19], y se compara su rendimiento con el de las arquitecturas de aprendizaje profundo abarcadas en el estudio.

III. Experimentación

La experimentación realizada en el estudio ofrece una vista de las diferentes perspectivas desde las que se puede abordar la tarea de detección facial y reconocimiento emocional de un usuario humano. Tradicionalmente, en el ámbito de la visión por computador, las principales estrategias utilizadas para el reconocimiento de objetos en imágenes han girado en torno al desarrollo de modelos matemáticos con los que inferir las cualidades de una imagen. El auge del paradigma del aprendizaje profundo ha proporcionado nuevas y potentes herramientas con las que inferir estas características de manera más exhaustiva a través de operaciones de convolución sobre la imagen. Si bien las técnicas enmarcadas en el ámbito del aprendizaje profundo habitualmente requieren de una capacidad de procesamiento elevada comparada con las técnicas tradicionales [20], el surgimiento de nuevas arquitecturas hardware especializadas en el cómputo tensorial, pilar fundamental del aprendizaje profundo, hacen viable la utilización de estas herramientas en ámbitos cotidianos.

En el estudio se presentan dos estrategias tradicionales y tres de vanguardia para la detección y clasificación de rostros en una imagen, con lo que se pretende mostrar una panorámica del alcance actual de estas técnicas y su aplicabilidad en hardware compacto. De manera concreta, se presentan dos arquitecturas alternativas para el reconocimiento emocional a partir de la expresividad facial humana: La utilización de un único modelo de aprendizaje profundo al que delegar la labor de detección y reconocimiento emocional de manera unificada, y la escisión de estas dos tareas en modelos separados. El motivo de esta comparativa es el evaluar la relación entre rendimiento y precisión de los sistemas derivados, estudiando si la combinación de modelos optimizados en la resolución de tareas atómicas presenta alguna ventaja en el ámbito del reconocimiento emocional frente a las técnicas de aprendizaje profundo.

A. *Métricas*

Desde la perspectiva de la evaluación del rendimiento de las arquitecturas propuestas para el estudio se han considerado las dimensiones de la precisión en detección de rostros, y la eficiencia computacional como principales medidores de calidad. Las métricas consideradas en cada caso son el tiempo de inferencia medio (1), entendiendo éste como el tiempo necesario para resolver la detección en una imagen cualquiera, la ratio de acierto medio (2) por imagen, referido al número de rostros identificados correctamente, y el ajuste de los cuadros resultantes de la detección a los esperados sobre los rostros. La métrica utilizada para cuantificar este último ajuste es la intersección sobre la unión (IoU) [21] que se define como el cociente entre el área de la región solapante entre dos recuadros dados y el área de su unión, como se representa en la ecuación (3). Las coordenadas consideradas en cada caso están normalizadas al tamaño de la imagen, con lo que se asegura que el IoU proporciona una medida acotada entre 0 y 1 de la bondad del ajuste entre el recuadro de detección obtenido durante inferencia y el control.

$$TIM = \frac{\sum_i^{Núm.imágenes} Tiempo\ de\ inferencia(i)}{Núm.imágenes} \quad (1)$$

$$RAM = \frac{\sum_i^{Núm.imágenes} \frac{detecciones(i)}{rostros(i)}}{Núm.imágenes} \quad (2)$$

$$IoU = \frac{Area\ Intersección}{Area\ Unión} \quad (3)$$

$$RA = \frac{\sum_i^{Núm.clases} Aciertos(i)}{\sum_i^{Núm.clases} Rostros(i)} \quad (4)$$

$$Acc = \frac{VN + VP}{VP + VN + FP + FN} \quad (5)$$

En lo relativo al rendimiento en el reconocimiento emocional sobre rostros acotados, la métrica utilizada es la ratio de aciertos (4) sobre cada una de las clases emocionales consideradas.

La precisión de los clasificadores se evalúa con (5) que nos indica la precisión en clasificación. Los términos VN, VP indican los verdadero positivos y verdaderos negativos de la matriz de confusión, mientras que FP y FN indican los falsos positivos y falsos negativos de la matriz de confusión.

B. *Entorno de experimentación*

Para el entrenamiento y la evaluación de los modelos generados durante la experimentación se ha recurrido a la plataforma Google Colab [22] para acceder a HW optimizado en el cómputo tensorial denominado TPU (*Tensor Processing Unit)*. En lo relativo al desarrollo de los modelos software, se ha empleado la API de Tensorflow v.1.15 [23] con motivo de maximizar el rendimiento con las arquitecturas del Tensorflow Detection Model Zoo [16] que se emplean en el estudio.

C. *Bases de datos*

El emplear un conjunto de datos adecuado durante la construcción de los modelos de detección juega un papel importante en el rendimiento final de estas arquitecturas. La forma en que se ofrece la información a los modelos durante el entrenamiento es tan importante como la arquitectura de los modelos en sí, de forma que la identificación de las bases de datos adecuadas y su posterior procesamiento es un punto fundamental del proceso de entrenamiento. Las bases de datos escogidas para el estudio se ajustan a los requisitos de cada configuración a evaluar.

Durante la experimentación relativa a la detección facial en tiempo real, se ha empleado la base de datos WIDER FACE [24], que contiene imágenes de rostros anotados en diversas poses, escalas y condiciones de iluminación, siendo una base de datos con imágenes altamente representativas de las condiciones esperadas en la operación en tiempo real. WIDER FACE presenta su base de datos dividida en tres subconjuntos de entrenamiento, validación y prueba, siendo estos los mismos conjuntos que se han empleado para este estudio.

Para la tarea de reconocimiento emocional de un rostro localizado se ha hecho uso de los conjuntos RAVDESS [25] y AffectNet [26], que incluyen extensas anotaciones sobre la expresividad facial de los rostros que contienen. RAVDESS presenta la información relativa a la expresividad facial en forma de vídeos en formato mp4 de entre uno y cuatro segundos de duración. Para poder utilizar esta base de datos en el estudio, se han extraído 20 fotogramas por instancia seleccionados de manera equidistante en cada vídeo, formando los conjuntos de entrenamiento, validación y prueba que se indican en la TABLA I. AffectNet por su parte cuenta con conjuntos de imágenes ya separados como se muestra en la TABLA I.

TABLA I: DIVISIÓN DE LAS BASES DE DATOS EN SUBCONJUNTOS DE ENTRENAMIENTO, VALIDACIÓN Y PRUEBA.

Base de datos	*Entrenamiento*	*Validación*	*Prueba*
WIDER[a]	75.0	12.5	12.5
RAVDESS[b]	70.0	20.0	10
AffectNet	98.7	0.65	0.65
ImageNet	86.0	4.0	10

[a.] El subconjunto de prueba no incluye anotaciones sobre los rostros. Este subconjunto se ignora. Las instancias de prueba utilizada se han extraído del subconjunto de validación original.

[b.] Distribución de los fotogramas extraídos de los vídeos.

D. *Arquitecturas evaluadas*

A razón de evaluar el rendimiento en detección y reconocimiento emocional a partir de rostros, y de estudiar al mismo tiempo las diferencias de rendimiento existentes entre

la utilización de modelos acoplados y desacoplados, se han preparado arquitecturas de diversa naturaleza atendiendo en todo caso al objetivo común de identificar correctamente la emoción experimentada por una persona a partir de sus rasgos faciales.

Las arquitecturas desacopladas consideradas están compuestas por dos módulos de operación principales, contando con un detector facial que identifica de manera preliminar los rostros en una imagen, y un reconocedor emocional, que toma como entrada el rostro acotado por el primer módulo y realiza una tarea de inferencia sobre el conjunto de emociones de interés. Pese a que en la actualidad las arquitecturas convolucionales basadas en aprendizaje profundo son capaces de realizar la tarea de detección facial y reconocimiento emocional en un solo estadio, se estima que el desacoplar ambas tareas para resolverlas en modelos independientes puede generar modelos optimizados para la resolución de las subtareas componentes, lo que en última instancia puede impactar positivamente en el rendimiento global del sistema.

En lo relativo a las arquitecturas utilizadas en la tarea desacoplada de detección facial, se han ajustado los modelos pre-entrenados de las arquitecturas SSD, RFCN y FasterRCNN como alternativas de vanguardia a la detección facial, empleando como optimizador la implementación del algoritmo de gradiente descendente con una razón de aprendizaje 0.01. Durante el ajuste de los modelos se han congelado los pesos de sus capas a excepción de la de clasificación, asumiendo que el conjunto de elementos para las que fueron entrenadas proporciona características suficientes para su clasificación. El tamaño del batch empleado en cada caso ha sido de 40 elementos, empleando 20 épocas de entrenamiento de 1500 pasos. Así mismo, se han evaluado implementaciones clásicas para la detección facial basadas en descriptores HOG y en el algoritmo Viola-Jones combinadas con máquinas de soporte vectorial como clasificadores, utilizando en este caso la implementación que ofrece [27]. Para la tarea desacoplada de reconocimiento emocional sobre rostros acotados se ha utilizado un modelo convolucional basado en la arquitectura VGG-16 pre-entrenada. Se congelan las capas de extracción de características y se reentrena con el del conjunto ImageNet, con una división entre subconjuntos como se muestra en la TABLA I. La optimización del modelo se ha llevado a cabo empleando el algoritmo Adam [28] con una razón de aprendizaje de 0.001, utilizando en este caso una configuración de 50 elementos por batch, 40 episodios de entrenamiento y 1000 pasos por época.

Las arquitecturas de detección facial se han ajustado con la base de datos WIDER, mientras que la red convolucional para el reconocimiento emocional ha empleado el set RAVDESS.

Para la evaluación de las alternativas acopladas a la detección facial y reconocimiento emocional se han ajustado las arquitecturas pre-entrenados SSD, RFCN y FasterRCNN a la tarea combinada de detección y evaluación de la expresividad facial, empleando en este caso las bases de datos de RAVDESS y AffectNet para el entrenamiento exclusivo de sus capas de clasificación.

IV. Resultados

Como parte de la evaluación del rendimiento y de la utilidad de los modelos de reconocimiento emocional propuestos en el ámbito de la ayuda a la conducción, las arquitecturas de inferencia resultantes se han implementado en los equipos empotrados Raspberry Pi 4B y NVIDIA Jetson Nano propuestos como sistemas compactos. Los algoritmos clásicos para la detección facial toman como premisa la optimización del cómputo durante esta inferencia, lo que se muestra en las evaluaciones obtenidas en sendos sistemas Raspberry y NVIDIA en la Fig. 1 presentándose las alternativas basadas en descriptores HOG y en el algoritmo VJ como los sistemas más eficientes en tiempo de ejecución, con 2.23 y 1.87 segundos por imagen en el caso de los sistemas HOG, y 0.81 y 0.68 segundos en los sistemas basados en VJ, ambas medidas sobre los equipos NVIDIA y Raspberry respectivamente. De manera opuesta, las arquitecturas con mayor deuda temporal se corresponden con los modelos Faster, con tiempos de ejecución medios de 18.78 y 16.54 segundos, y con el modelo basado en la arquitectura RFCN, con tiempos de 29.65 y 24.68 segundos de ejecución en los equipos NVIDIA y Raspberry respectivamente. Especial atención merece la arquitectura SSD, que enmarcándose en el conjunto de técnicas de aprendizaje profundo ha mostrado unos tiempos de ejecución cercanos a las técnicas clásicas con una media de 4.27 y 3.61 segundos de ejecución por imagen en los sistemas NVIDIA y Raspberry.

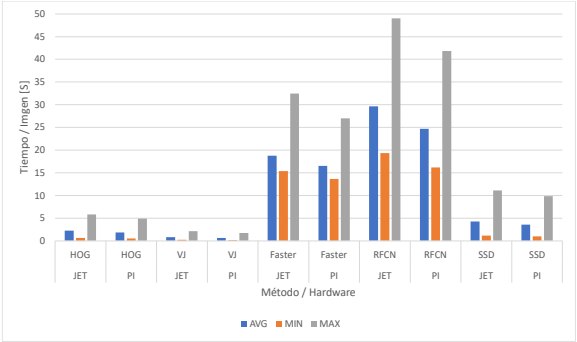


Fig. 1. Tiempo de procesamiento por imagen con los algoritmos HOG, Viola Jones (VJ), arquitectura Faster R-CNN (Faster), RFCNN y SSD en los sistemas empotrados de Jetson (JET) y Raspberry pi (PI).

En lo relativo a la evaluación de la precisión, como se muestra en la Fig. 2 el ratio de acierto medido entre los diferentes modelos implementados muestra un decaimiento en las técnicas de detección clásicas, con un ratio de 0.38 como peor valor medio en el caso de la arquitectura basada en VJ, y un mejor ratio medio global de 0.73 en el caso de las arquitecturas SSD.

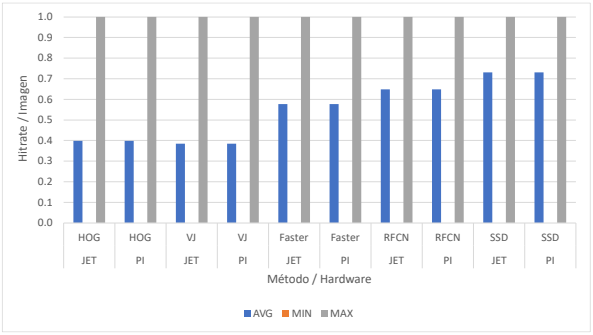


Fig. 2. Ratio de acierto de los algoritmos HOG, Viola Jones (VJ), arquitectura Faster R-CNN (Faster), RFCNN y SSD en los sistemas empotrados de Jetson (JET) y Raspberry pi (PI).

La ratio IoU proporciona una medida de la bondad del ajuste de la ventana de detección, obtenida sobre cada rostro, frente al esperado. Esta medida muestra de manera numérica la discrepancia que existe en una detección cualquiera en relación a la región real en que se encuentra el objeto a detectar, de forma que un valor próximo a 1 implica un ajuste bueno, y un valor cercano a 0 implica un ajuste nulo. Los resultados obtenidos en la experimentación que se facilitan en la Fig. 3 demuestran que los modelos de detección clásicos ofrecen un ajuste notablemente peor en términos absolutos en comparación a las alternativas modernas, con una ratio de 0.17 en el caso de las alternativas basadas en descriptores HOG frente al ratio de 0.42 que muestra la arquitectura SSD. Estos resultados medios se han computado sobre todas las detecciones obtenidas en cada caso, sin embargo, si se establece un límite inferior correspondiente al valor de ajuste mínimo necesario para considerar que una detección es un acierto, estableciendo este valor en un punto medio de la métrica (0.5), se observa que tanto los modelos clásicos como los modernos tienen valores de ajuste similares con ratios IoU de 0.69. Este resultado implica que las ventanas de detección que se corresponden con un acierto presentan un ajuste similar en todas las arquitecturas.

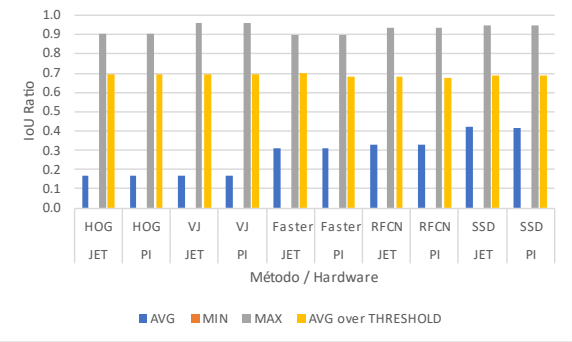


Fig. 3. Ratio IoU entre las áreas predichas y reales con los algoritmos HOG, Viola Jones (VJ), arquitectura Faster R-CNN (Faster), RFCNN y SSD en los sistemas empotrados de Jetson (JET) y Raspberry pi (PI).

En lo relativo al rendimiento de los sistemas propuestos en la tarea de reconocimiento emocional, la red convolucional basada en la arquitectura VGG-16 que se emplea en la tarea desacoplada de reconocimiento emocional sobre rostros acotados muestra una precisión (5) media en clasificación del 90% sobre todas las etiquetas emocionales consideradas en el la base de datos RAVDESS, como se muestra en la matriz de confusión de la Fig. 4. Por otra parte, las arquitecturas alternativas acopladas para la detección facial y reconocimiento emocional combinado han retornado valores de precisión en clasificación considerablemente peores a la alternativa desacoplada, presentando la arquitectura SSD con mejores resultados una media de aciertos del 39.15% en la base de datos de RAVDESS, como se muestra en la matriz de la Fig. 5.

	Neutral	Calm	Happy	Sad	Angry	Fearful	Disgust	Surprised
Neutral	89.17	1.92	1.88	1.51	1.70	1.53	0.91	1.33
Calm	1.99	88.66	1.74	1.52	1.45	1.69	1.94	1.01
Happy	1.65	1.82	89.65	0.96	1.23	1.55	1.17	1.97
Sad	1.20	1.54	1.10	90.36	1.98	1.60	1.26	0.96
Angry	1.68	1.25	1.03	1.47	89.86	1.73	1.29	1.49
Fearful	1.42	1.68	1.03	1.94	1.89	88.99	1.76	1.29
Disgust	1.31	1.16	1.30	1.21	1.01	1.12	91.72	1.17
Surprised	1.20	1.09	1.37	1.11	0.94	1.00	1.66	91.63

Fig. 4. Matriz de confusión obtenida con la arquitectura VGG-16 y a la base de datos RAVDESS.

	Neutral	Calm	Happy	Sad	Angry	Fearful	Disgust	Surprised
Neutral	39.24	9.25	8.62	8.98	8.45	8.12	7.66	9.68
Calm	8.96	38.92	9.12	9.03	9.74	7.66	8.32	8.25
Happy	9.85	9.66	39.82	7.52	8.12	8.79	7.33	8.91
Sad	8.60	9.88	9.03	40.01	9.12	7.78	8.20	7.38
Angry	7.99	7.80	9.97	8.53	39.02	8.10	9.02	9.56
Fearful	8.54	9.03	8.81	9.96	8.63	37.99	8.01	8.94
Disgust	9.63	9.55	8.25	8.97	8.00	7.21	39.21	9.18
Surprised	8.72	8.01	9.27	9.13	7.84	8.33	9.73	38.97

Fig. 5. Matriz de confusión obtenida con la arquitectura SSD y la base de datos RAVDESS.

V. Conclusiones y Trabajos Futuros

De manera general, los resultados obtenidos sugieren una mayor precisión de las arquitecturas modernas basadas en aprendizaje profundo frente a las alternativas clásicas, acarreando un sobrecoste computacional frente a éstas. Los resultados excepcionalmente pobres obtenidos en los modelos de reconocimiento emocional acoplados, por otra parte, indican un pobre aprendizaje en las arquitecturas implicadas de las características expresivas más significativas en la tarea de reconocimiento emocional; los modelos propuestos son capaces de identificar correctamente el rostro en la imagen, sin embargo, la inferencia de la etiqueta emocional resulta problemática. Si bien estos resultados pueden ser consecuencia de un aprendizaje insuficiente de las redes propuestas, tanto en lo relativo a la naturaleza de las bases de datos utilizadas como a las limitaciones inherentes derivadas del entrenamiento parcial con capas congeladas de las arquitecturas convolucionales, especialmente de las SSD, la alternativa al reconocimiento emocional desacoplada ofrece resultados de precisión notablemente superiores habiéndose entrenado con los mismos conjuntos de datos.

Desde la perspectiva de la eficiencia computacional las técnicas clásicas basadas en clasificadores automáticos sencillos muestran un mejor desempeño general frente a sus contrincantes modernas, lo que continúa legitimando su utilización en entornos en los que se prime la eficiencia sobre la precisión.

En lo relativo al rendimiento de los sistemas empotrados empleados en el estudio, tanto el sistema Raspberry como NVIDIA han mostrado tiempos de ejecución similares durante la etapa de evaluación, si bien el sistema NVIDIA Jetson disponible debería mostrar un rendimiento superior a su contrincante en las tareas de inferencia que involucrasen operaciones tensoriales, las imágenes de las bases de datos empleadas en la experimentación han llegado a saturar la memoria del dispositivo durante la evaluación, dificultando la tarea de extracción de resultados significativos.

De esta forma, los siguientes pasos en el estudio propuesto giran en torno al ajuste fino de las arquitecturas de reconocimiento emocional en los dispositivos empotrados y su posterior evaluación en un entorno de conducción real. De manera adicional, en vista de los pobres resultados en la tarea de reconocimiento emocional retornados por las arquitecturas basadas en aprendizaje profundo, se estima oportuno hacer

una nueva iteración en la fase de entrenamiento con la que poder profundizar en las causas de dicha discrepancia en el rendimiento frente a la alternativa de detección y reconocimiento emocional desacoplada.

La experimentación propuesta muestra que, desde la perspectiva de la eficiencia computacional, las estrategias clásicas para la detección de objetos en imágenes siguen teniendo cabida en los sistemas actuales. Por otra parte, las arquitecturas modernas basadas en aprendizaje profundo proporcionan resultados razonablemente eficientes, como es el caso del modelo SSD expuesto en el estudio. Actualmente el uso de arquitecturas de aprendizaje profundo está muy extendido en el ámbito de la visión por computador; si bien de manera general estas técnicas ofrecen resultados satisfactorios bajo las tareas de detección y clasificación de objetos en imágenes planas, su combinación con otras técnicas clásicas de procesamiento de imagen puede aportar beneficios constatables en el rendimiento final del sistema integrador. El seguir analizando el potencial de esta simbiosis de técnicas es, en última instancia, un objetivo permanente a perseguir en estudios futuros.

REFERENCIAS

[1] "Accidentes de Tráfico con víctimas - Dataset," *datos.gob.es*, 2014. .

[2] A. Gutierrez, "El estrés aumenta un 28% el riesgo de accidente," *Revista DGT*, 2017.

[3] "Psicología aplicada a la conducción," *Dirección General de Tráfico*, 2017.

[4] J. A. Groeger, *Understanding Driving*. Routledge, 2013.

[5] C. Castro, *Human Factors of Visual and Cognitive Performance in Driving*. CRC Press, 2008.

[6] Ecomotor.es, "Sistemas ADAS: qué son, cómo funcionan y por qué la DGT los considera tan importantes para la seguridad," *El Econ.*, 2018.

[7] "Los vehículos para los exámenes, con ADAS," *Revista DGT*, 2019. .

[8] J. S. K. Ooi, S. A. Ahmad, Y. Z. Chong, S. H. M. Ali, G. Ai, and H. Wagatsuma, "Driver emotion recognition framework based on electrodermal activity measurements during simulated driving conditions," in *IECBES 2016 - IEEE-EMBS Conference on Biomedical Engineering and Sciences*, 2016, pp. 365–369, doi: 10.1109/IECBES.2016.7843475.

[9] X. Meng, K. K. Lee, and Y. Xu, "Human driving behavior recognition based on hidden Markov models," in *2006 IEEE International Conference on Robotics and Biomimetics, ROBIO 2006*, 2006, pp. 274–279, doi: 10.1109/ROBIO.2006.340166.

[10] M. Jabon, J. Bailenson, E. Pontikakis, L. Takayama, and C. Nass, "Facial-expression analysis for predicting unsafe driving behavior," *IEEE Pervasive Comput.*, vol. 10, no. 4, pp. 84–95, Oct. 2011, doi: 10.1109/MPRV.2010.46.

[11] "Raspberry Pi 4 Model B," *Raspberry Pi*, 2020. .

[12] "New Jetson Nano 2GB Developer Kit," *NVIDIA*. .

[13] W. Liu *et al.*, "SSD: Single shot multibox detector," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2016, vol. 9905 LNCS, pp. 21–37, doi: 10.1007/978-3-319-46448-0_2.

[14] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object detection via region-based fully convolutional networks," in *Advances in Neural Information Processing Systems*, 2016, pp. 379–387, doi: 10.5555/3157096.3157139.

[15] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, 2017, doi: 10.1109/TPAMI.2016.2577031.

[16] Y. Shi, "TensorFlow 1 Detection Model Zoo," *GitHub*, 2020. .

[17] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2001, vol. 1, doi: 10.1109/cvpr.2001.990517.

[18] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proceedings - 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005*, 2005, vol. I, pp. 886–893, doi: 10.1109/CVPR.2005.177.

[19] M. ul Hassan, "VGG16 - Convolutional Network for Classification and Detection," *Neurohive*, 2018. .

[20] N. K. Chauhan and K. Singh, "A Review on Conventional Machine Learning vs Deep Learning," in *2018 International Conference on Computing, Power and Communication Technologies (GUCON)*, 2018, pp. 347–352, doi: 10.1109/GUCON.2018.8675097.

[21] Adrian Rosebrock, "Intersection over Union (IoU) for object detection," *PyImageSearch*, 2016. .

[22] G. Inc., "Te damos la bienvenida a Colaboratory," 2021. .

[23] G. Inc., "TensorFlow Core v1.15.0," 2020. .

[24] S. Yang, P. Luo, C. C. Loy, and X. Tang, "WIDER FACE: A face detection benchmark," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016, vol. 2016-Decem, pp. 5525–5533, doi: 10.1109/CVPR.2016.596.

[25] S. Livingstone and F. Russo, "The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS)," *PLoS One*, vol. 13, Apr. 2018, doi: 10.5281/zenodo.1188976.

[26] A. Mollahosseini, B. Hasani, and M. H. Mahoor, "AffectNet: A Database for Facial Expression, Valence, and Arousal Computing in the Wild," *IEEE Trans. Affect. Comput.*, vol. 10, no. 1, pp. 18–31, Jan. 2019, doi: 10.1109/TAFFC.2017.2740923.

[27] A. Zihna and C. Lorentzen, "Scikit-Learn: Machine Learning in Python," *GitHub*, 2021. .

[28] "Optimizers," *Keras API*. .

XIX Conferencia de la Asociación Española para la Inteligencia Artificial

I Taller de Inteligencia Artificial para el Desarrollo (AI4D)



Teledección híbrida en cultivos leñosos con codificador automático convolucional

1st Iván Rico Prieto
dept. I+D UTW-Agro
Unmanned Technical Works
Madrid, España
ivan.rico@utw.es

2rd Roberto Hermoso Peralo
dept. I+D UTW-Agro
Unmanned Technical Works
Madrid, España
roberto.hermoso@utw.es

3th Luz Karime Atencia Payares
dept. I+D UTW-Agro
Unmanned Technical Works
Madrid, España
luz.atencia@utw.es

4nd Jesús Cano Jiménez
dept. I+D UTW-Agro
Unmanned Technical Works
Madrid, España
jesus.cano@utw.es

***Resumen*—La teledetección es una técnica de análisis ampliamente utilizada en la agricultura de precisión. Existen principalmente dos tipos de fuentes de información, las de alta resolución (típicamente imágenes tomadas con aeronaves no tripuladas) y las de baja resolución (típicamente imágenes satelitales). La primera de estas presenta una gran resolución espacial, mientras que la segunda presenta una buena resolución temporal. Nuestro objetivo ha sido combinar estos dos tipos de información para conseguir imágenes de alta resolución temporal y espacial. Para ello, creamos un modelo de red neuronal convolucional con una arquitectura basada en un codificador automático, el cual entrenamos con información recopilada durante los últimos años. El resultado obtenido es que la precisión de nuestro modelo es suficiente buena como para realizar los análisis agronómicos pertinentes.**

I. Introducción

El estudio de cultivos leñosos (olivo, frutales, vid) a través de la teledetección, requiere resoluciones mayores que el estudio de cultivos herbáceos, ya que es conveniente poder distinguir el propio cultivo de la cubierta vegetal, la capa de vegetación que cubre la superficie no productiva.

Actualmente la organización que ofrece mejor resolución en las bandas multiespectrales (necesarias para un estudio agronómico) de manera gratuita, abierta y periódica es la ESA (Agencia Espacial Europea), a través de la misión Sentinel-2, que cuenta con dos satélites. Estos satélites proporcionan información con una resolución espacial de 10 m/píxel (que no es suficiente para poder distinguir los diferentes elementos de un cultivo heterogéneo) y una resolución temporal de 2-3 días.

Por otro lado, la manera de obtener esta información multiespectral con la mayor resolución espacial posible es mediante el uso de aeronaves no tripuladas equipadas con cámaras adecuadas, alcanzando resoluciones espaciales de 10 cm/píxel. Como desventaja, estos servicios no son gratuitos con lo que la resolución temporal suele ser baja.

La cuestión que nos planteamos es: ¿pueden las técnicas de aprendizaje supervisado servir para combinar estas dos fuentes de información ofreciendo una tercera alternativa con buena resolución tanto espacial como temporal?

La hipótesis que planteamos es que, para un cultivo leñoso, en el que los elementos de interés son fijos (la posición de los árboles es fija), partiendo de suficiente información de alta resolución de un mismo cultivo, un modelo puede aprender el patrón con el que se desarrollan los elementos que intervienen en dicho cultivo (árboles y cubierta vegetal). De este modo, teniendo una única imagen de alta resolución (que contiene la información espacial relevante) y una imagen de baja resolución de una fecha diferente (posterior), el modelo podrá determinar cuál es la manera más probable en la que habrá evolucionado el cultivo (a nivel de alta resolución) para que los valores promedio correspondan con la imagen de baja resolución.

La cuestión planteada y la hipótesis realizada trazan la línea que tomamos para abordar esta cuestión, pues la idea de utilizar técnicas de aprendizaje supervisado sugiere el uso de redes neuronales, el tratamiento de imágenes sugiere concretamente el uso de redes neuronales convolucionales y la necesidad de estimar una evolución del cultivo sugiere emplear una estructura de modelo generativo.

II. Materiales y métodos

II-A. Descripción de los datos disponibles

Existen numerosos índices vegetativos para estudiar un cultivo, en esta ocasión nos centramos en el NDVI (Índice de Vegetación de Diferencia Normalizada) por su extendido y exitoso uso, dejando la puerta abierta a ampliar el estudio a otros índices de interés aplicando técnicas similares junto con técnicas de aprendizaje por transferencia.

La información de la que partimos para entrenar nuestro modelo predictivo son las imágenes satelitales de la misión Sentinel-2 (que como ya hemos indicado, son abiertas y gratuitas) y las imágenes tomadas con una aeronave no tripulada dotada de una cámara multiespectral; concretamente la aeronave “Sensefly eBee X” y la cámara “Parrot Sequoia”. Como cabe esperar, el factor limitante (lo que determinará nuestro volumen de datos) es la cantidad de información de alta resolución que tenemos, en concreto contamos con 30 vuelos sobre una finca experimental ubicada en Aranjuez, espaciados homogéneamente en el tiempo durante las campañas de 2017, 2018, 2019 y 2020 y con 11 vuelos sobre una una finca de producción ubicada en Toledo, espaciados durante las campañas de 2017, 2018 y 2019. Ambas fincas están cultivadas con olivar y tienen una cubierta vegetal espontánea. La finca experimental tiene 9 hectáreas de superficie y está

cultivada con Cornicabra y Picual en intensivo y Arbequina en superintensivo, mientras que la finca de producción tiene 18 hectáreas y está cultivada con Arbequina en superintensivo. En ambas fincas hay zonas con árboles de diferentes edades.

A la vista de la información que disponemos, restringimos este estudio a cultivos intensivos y superintensivos de olivar con cubierta vegetal espontánea, dejando nuevamente la puerta abierta a extender el estudio a otros tipos de cultivo aplicando técnicas similares junto con técnicas de aprendizaje por transferencia.

II-B. Preparación de los datos

Por razones de practicidad y simplificación, redimensionamos las imágenes tomadas con la aeronave (de unos 10 cm/píxel) a 1 m/píxel, ya que esta resolución ya es adecuada para realizar los estudios agronómicos pertinentes y supone una gran simplificación para el modelo. De este modo, en lo que sigue llamaremos baja resolución a 10 m/píxel y alta resolución a 1 m/píxel.

Por otro lado, en principio puede parecer que 41 imágenes de alta resolución con sus respectivas 41 imágenes de baja resolución (de la misma fecha) no pueden ser suficientes para entrenar una red neuronal, pero realmente, cada imagen abarca una superficie muy extensa y contiene mucha información. El mecanismo que hemos seguido ha sido tratar cada imagen como si realmente fuera un conglomerado de muchas imágenes más pequeñas, a las que llamamos "imágenes mínimas", de modo que el modelo trabaje a nivel de imágenes mínimas y no de imágenes. Esta táctica, además, es muy conveniente para tratar con parcelas de diferentes tamaños, ya que cualquier imagen previamente será dividida en imágenes mínimas, las cuales serán procesadas por el modelo y posteriormente serán recompuestas para formar la imagen final.

Tenemos dos tipos de imagen mínima, las de alta y baja resolución. Ambas representan la misma superficie. La elección de imagen mínima más simple que se puede hacer es la que corresponde con un píxel de baja resolución, es decir, la imagen mínima de baja resolución sería un único píxel (con un único valor) y la imagen mínima de alta resolución sería de 10x10 píxeles, ambas representando una superficie de 100m^2 en el terreno:

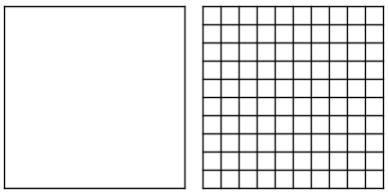


Figura 1. Imágenes mínimas de baja y alta resolución más simples.

La siguiente elección de imagen mínima más sencilla es aquella que corresponde con 4 píxeles de baja resolución y por tanto 40x40 píxeles de alta resolución, representando por tanto 400m^2 en el terreno. Esta elección presenta una ventaja frente a la anterior, y es que permite fraccionar una imagen en imágenes mínimas con solape, y eso sirve para eliminar los factores de borde. Por tanto, esta será nuestra elección de formato para la imagen mínima.

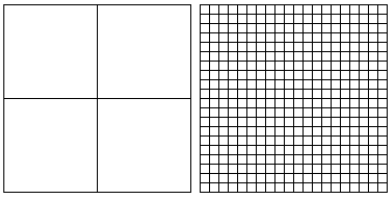


Figura 2. Imágenes mínimas de baja y alta resolución con posibilidad de solape.

Para justificar la preparación de los datos, conviene describir desde un punto de vista externo de qué modo funciona el modelo. Partimos de información de dos fechas diferentes (llamémoslas fecha1 y fecha2):

- Una imagen mínima de alta resolución de fecha1.
- Una imagen mínima de baja resolución de fecha1.
- Una imagen mínima de baja resolución de fecha2.

Queremos que el modelo nos devuelva la imagen mínima de alta resolución de fecha2. Por tanto, queremos que el modelo extraiga la información espacial de la imagen de alta resolución de fecha1 y que la perturbe en función de los cambios medios temporales entre fecha2 y fecha1. Por esta razón, podemos simplificarlo aun más sin eliminar información introduciendo al modelo la imagen mínima de alta resolución de fecha1 y la diferencia de las imágenes de baja resolución de fecha2 y fecha1, cuyos valores representarán los factores medios con los que crece o decrece la superficie vegetal en el terreno representado por las imágenes mínimas.

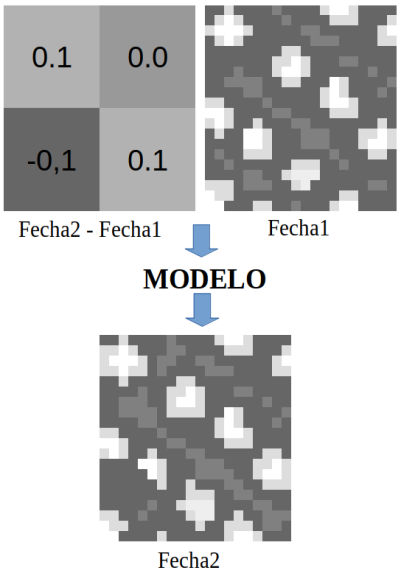


Figura 3. Ejemplo de la entrada y salida de datos del modelo.

En la Figura 3 puede verse un ejemplo de la entrada y salida del modelo. En las imágenes mínimas de alta resolución, los

tonos claros representan zonas con alta densidad de vegetación y la imagen de baja resolución (cuyo valor es fecha2 - fecha1), representa una serie de coeficientes, que en caso de ser negativos indican que en esa zona ha desaparecido vegetación y en caso de ser positivos indican que ha crecido vegetación.

De este modo, el conjunto de datos de entrenamiento y validación estará formado por todas las combinaciones de fechas que podamos hacer para cada imagen mínima que podamos formar. Realizando algunas restricciones necesarias (como por ejemplo que fecha2 sea posterior a fecha1 y que ambas fechas pertenezcan a la misma campaña) el volumen de datos que tenemos es de unos 420000 grupos de datos como los mostrados en la Figura 3. De estos datos, separamos de forma aleatoria el 20 % para validación del modelo y dejamos el 80 % restante para entrenamiento.

II-C. Descripción del modelo

Dado que la entrada y la salida del modelo son imágenes, es razonable que dicho modelo sea una red neuronal convolucional, cuya entrada es un tensor tridimensional que contiene los valores de la imagen mínima de alta resolución de fecha1 y los valores de la diferencia de las imágenes mínimas de baja resolución de fecha2 y fecha1, y cuya salida es un tensor bidimensional (matriz) con los valores de imagen mínima de alta resolución de fecha2.

Como queremos que el modelo sea capaz de generar o eliminar superficie vegetal, es razonable escoger una estructura inspirada en algún modelo generativo. La arquitectura que hemos escogido como referencia para crear nuestro modelo es la de un codificador automático convolucional (para más información véase [1]). Este modelo tiene a su vez dos subestructuras, un codificador, que consiste en una serie sucesiva de convoluciones que condensan la información del tensor de entrada en un menor número de variables, y un decodificador, que consiste en una serie de convoluciones transpuestas (o deconvoluciones) sucesivas que a partir de la información compactada del codificador, son capaces de reconstruir un tensor parecido al de entrada.

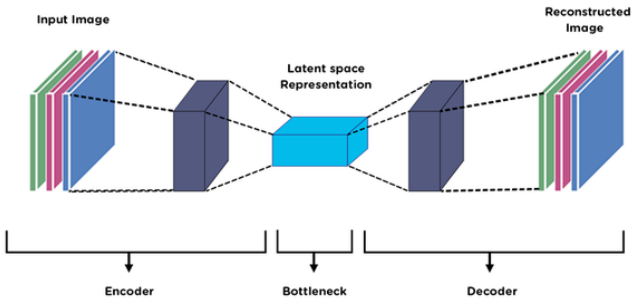


Figura 4. Estructura de un codificador automático.

Este tipo de redes ha sido usado para multitud de fines, como eliminar ruido de imágenes médicas, segmentación de imágenes, y por supuesto, generación de imágenes (véase [2] y [3]).

III. Resultados

Estudiando la evolución de la función pérdida (error cuadrático medio) durante el entrenamiento, podemos apreciar que el modelo está aprendiendo el patrón con el que evoluciona el olivar y su cubierta vegetal, ya que en cada época el error es más pequeño, tanto con los datos de entrenamiento como con los datos de validación.

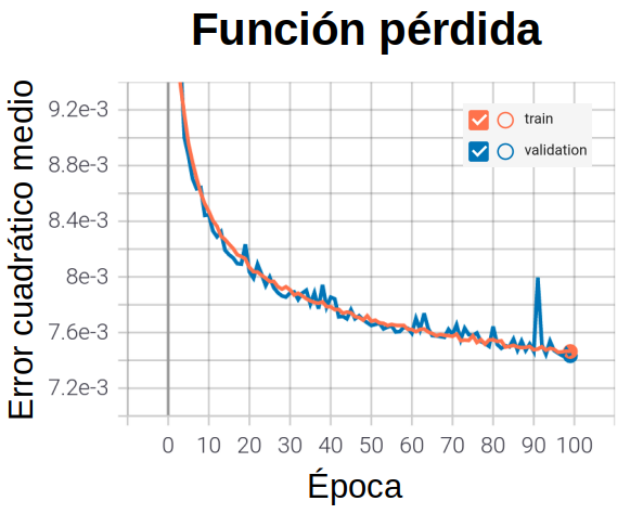


Figura 5. Evolución de la función pérdida durante el entrenamiento.

A partir de la gráfica, puede apreciarse que no hay sobreajuste, ya que tanto la función pérdida aplicada a los datos de entrenamiento como la función pérdida aplicada a los datos de validación toman aproximadamente los mismos valores en cada época. Además, aparentemente el modelo puede seguir aprendiendo, ya que la función pérdida, incluso en las últimas épocas, presenta pendiente negativa. No obstante, no hemos aumentado el número de épocas ya que esta observación es un efecto visual debido a las escalas de la gráfica, realmente el aprendizaje significativo ocurre hasta la época 20 aproximadamente. En definitiva, la conclusión importante que obtenemos es que el modelo ha aprendido ya que la función pérdida tiene pendiente negativa en todo momento y que no presenta sobreajuste.

Para tratar de evaluar el modelo en un caso límite, separamos dos imágenes de alta resolución que no han sido utilizadas ni como datos de entrenamiento ni como datos de validación. Además escogemos estas imágenes en los momentos en los que hay más variación de vegetación: una de las imágenes es de finales de abril, que es cuando la cubierta vegetal está más desarrollada, mientras que la otra imagen es de principios de julio, momento en el que prácticamente toda la cubierta se ha secado y la única superficie vegetal que hay son los olivares. Entonces, a partir del modelo ya entrenado, tratamos de predecir la imagen de alta resolución de julio a partir las imágenes del alta y baja resolución de abril y la imagen de

baja resolución de julio. Después comparamos el resultado del modelo con el resultado real.

Haciendo la comparativa entre las imágenes de alta resolución de abril y julio, para confirmar que efectivamente son diferentes, obtenemos el siguiente resultado:

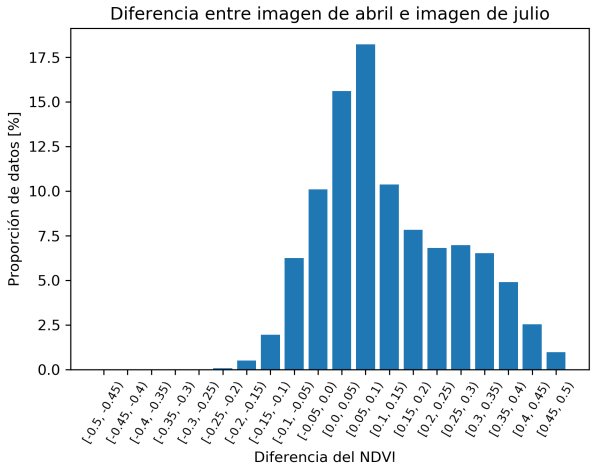


Figura 6. Diferencia del NDVI entre abril y julio.

Podemos apreciar que la distribución de diferencias del índice vegetativo está claramente desviado hacia valores positivos, lo cual es razonable ya que en abril la cubierta vegetal está mucho más desarrollada y en julio está prácticamente seca.

Realizando la misma comparativa entre la imagen real de alta resolución de julio y la imagen calculada de alta resolución para esta misma fecha, obtenemos el siguiente resultado:

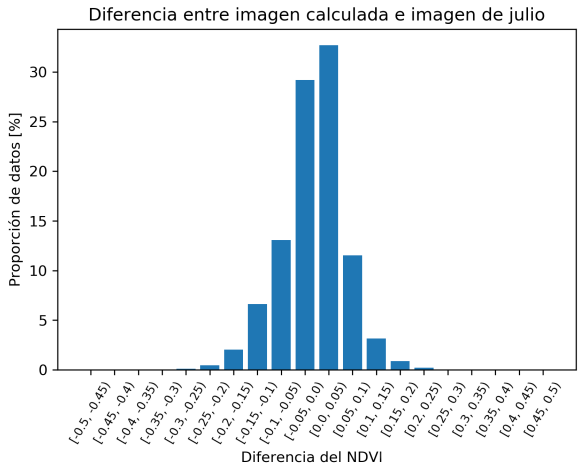


Figura 7. Diferencia del NDVI entre la imagen real de julio y la imagen calculada para esta misma fecha.

Podemos apreciar que ambas imágenes son muy parecidas, ya que siguen una distribución normal centrada en el origen. Además podemos ver que el NDVI de más del 60 % de los datos difiere en menos de 0,05 y que aproximadamente el 90 % de los datos difiere en menos de 0,1. Estas diferencias son perfectamente válidas para el fin que persiguen estos mapas. Idealmente, la imagen calculada debería ser exactamente igual que la imagen real, pero dado que no todos los elementos de interés tienen posición fija (los árboles sí tienen posición fija pero la cubierta vegetal no), es razonable que la diferencia de estas dos imágenes presente una dispersión que siga una distribución normal.

Para mostrar visualmente estos resultados, hemos seleccionado la misma sección de parcela, que incluye caminos y varios tipos de cultivo, de las imágenes reales de alta resolución de abril y julio y la imagen de alta resolución calculada para julio.

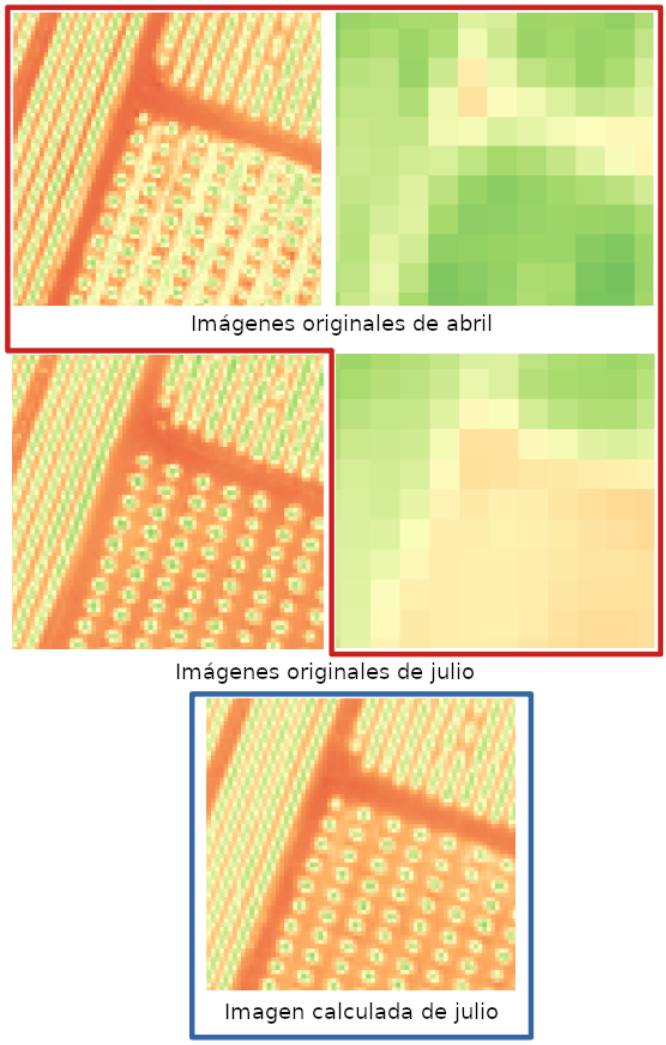


Figura 8. Comparativa entre imágenes de alta y baja resolución reales e imagen de alta resolución calculada.

Recuadrado en rojo, se encuentra la información que entra en el modelo (realmente la entrada es la imagen de alta resolución de abril y la diferencia entre las imágenes de baja resolución de julio y abril) y en recuadrado en azul, la salida del modelo. Podemos apreciar cómo el modelo ha eliminado la

cubierta vegetal del cultivo intensivo, obteniendo un resultado prácticamente idéntico a la imagen de alta resolución real.

IV. Conclusión

A partir de los resultados, concluimos que el objetivo de este estudio se ha logrado exitosamente: partiendo de este modelo entrenado y de una sola imagen de alta resolución por campaña de un cultivo de olivar intensivo o superintensivo con cubierta vegetal espontánea y partiendo de las imágenes gratuitas y periódicas del Sentinel-2, podemos obtener imágenes de alta resolución espacial con la misma resolución temporal del Sentinel, de 2 a 3 días.

Hay que tener en cuenta también los sesgos de información y las limitaciones que tenemos, por ejemplo, los datos con los que hemos entrenado el modelo son de fincas de olivar con diferentes sistemas de producción y variedades, pero ambas están ubicadas en la zona central de España (Madrid-Toledo), y seguramente al aplicar el modelo a una finca de olivar ubicada en un lugar con un clima diferente, los resultados no sean tan buenos, ya que la cubierta vegetal y el propio olivar posiblemente evolucionen de una forma ligeramente diferente.

A pesar de estas limitaciones, lo que aquí presentamos es un método para combinar las ventajas de las imágenes satelitales (resolución temporal) y las ventajas de las imágenes obtenidas con aeronaves no tripuladas (resolución espacial), ampliable a otros cultivos y a otros índices de interés.

REFERENCIAS

[1] Ian Goodfellow and Yoshua Bengio and Aaron Courville, "Deep Learning". Chap.14. 2026.
[2] Lee D, Yu H, Jiang X, Rogith D, Gudala M, Tejani M, Zhang Q, Xiong L. Generating sequential electronic health records using dual adversarial autoencoder. J Am Med Inform Assoc. 2020 Jul 1;27(9):1411-1419. doi: 10.1093/jamia/ocaa119. PMID: 32989459; PMCID: PMC7647348.
[3] arXiv:1902.05581 [cs.LG]

Decision Support System for Agricultural Production in Sub-Saharan Africa

Bradwell J Garanganga, Stand 14220, Leggate Ave, Borrowdale, HARARE, Zimbabwe. Mobile:-+263-772220330 (also What's App accessible) Email: bjgaranganga@gmail.com

Trymore Nyakutambwa: Digitron Business Solutions, 158 Herbert Chitepo Avenue, HARARE, Zimbabwe Tel: +263-772305436, Email: trymoren@digitron@co.zw .

James Murombedzi, ACPC/ UN Economic Commission for Africa, Menelik II Ave., P.O. Box 3001, ADDIS ABABA, Ethiopia, Tel: 251-11-544-5000,Fax: 251-11-551 4416 (Addis Ababa), Email: murombedzi@un.org

Frank Rutabingwa, ACPC/ UN Economic Commission for Africa, Menelik II Ave., P.O. Box 3001, ADDIS ABABA, Ethiopia, Tel: 251-11-544-5000,Fax: 251-11-551 4416 (Addis Ababa) Email: frutabingwa@un.org

ABSTRACT

Many countries in Sub-Saharan Africa experience food security challenges from to time mainly due to the vagaries of weather and climate. However, taking advantage of advances in climate sciences, in particular seasonal climate forecasting, agricultural productivity can be enhanced, especially using crop yield prediction models. Notwithstanding this, there are data gaps and yield forecast gaps that need to be filled in order for farmers to improve their decision making. Advances in Artificial Intelligence could be exploited to address these and help improve the decision support systems for agricultural production. These systems can be put to good uses by the farmers in order to plan their agricultural seasons much better resulting in more efficient productivity.

I. INTRODUCTION

Environmental conditions can either support or negatively affect agricultural productivity. Climate, especially its variability, is one such environmental control on many facets of agricultural production systems. For instance, it is well-known that the quality and quantity of parameters such as temperatures, rainfall, wind, radiation, all have varying degrees of impacts on crops and livestock production. Therefore, provision of advisories on monitoring and prediction of performances of these parameters can go a long way in sustainable and climate smart agriculture. The study demonstrates the beneficial use of seasonal climate forecasts (SCF) in the prediction of crop yields for the food security and agricultural sector. Challenges resulting from data gaps can be addressed through Artificial Intelligence (AI) in order to improve SCF and in turn crop yield prediction in SSA.

II. CLIMATE APPLICATION IN AGRICULTURAL PRODUCTION SYSTEMS

The World Bank Group estimates that with a current hydrometeorological investment portfolio approximated at US$ 500 million that globally improved weather, climate, as well as water observation and forecasting could lead to up to US$ 30 billion per year in increases in global productivity and up to US$ 2 billion per year in reduced asset losses, Hallegatte (2012). This scale of improved productivity could be crucial to lifting out of poverty the millions around the world whose livelihoods are at risk of climate shocks. It has been demonstrated that investments towards capacitating the institutions mandated for the production and application of Climate Information Systems (CIS) has potential to yield 10 times in returns of such investment, UNECA (2020). Agriculture is a major beneficiary of a well-developed CIS. Therefore, there needs to be intensified efforts in order to upscale the utilization of climate information in crop production across countries in the Sub-Saharan Africa (SSA).

III. TECHNIQUES IN APPLICATION OF CLIMATE INFORMATION AS DECISION SUPPORT

The past several decades have witnessed phenomenal advances in the understanding of the laws that govern the characteristics and behaviour of the global climate system. This has helped in the development of suites of general circulation ocean-atmosphere models: numerical weather prediction models, regional climate models which resulted in improvement in prediction skills, accuracy and reliability of multi-timescale (short- and medium-range, sub-seasonal, and seasonal and inter-annual) forecast products. Seasonal climate forecasts range from the next three to six months. These forecasts can be of great utility for a number of climate sensitive sectors (e.g. agriculture, disaster management, health, energy and water resource management). Application of CIS has been demonstrated to help in better planning of multi-sectoral socioeconomic development yielding enormous benefits to society, Coelho and Costa (2010).

i. Seasonal climate forecasting techniques

There are two basic approaches to seasonal climate forecasting: dynamical and statistical techniques. The former requires enormous resources, in particular with respect to computing facilities. This has seen concentration of research and development of SCF products in the more advanced economies. The statistical techniques can be effective tools and many national Meteorological and Hydrological Services in SSA are able to generate seasonal forecast using modest computing resources. Dynamical coupled ocean-atmosphere general circulation models promise to extend the accuracy and lead time with which seasonal rainfall forecasts can be made beyond that of pure statistical schemes. This is because non-stationary climate purely statistical schemes based on observed relationships between climate indicators and local variables will from time to time fail when climate drifts away from the regime for which the model was initially tuned. In practice, however, the best statistical models (for example the current Bureau of Meteorology seasonal climate outlook scheme (Drosdowsky and Chambers: 2001) still out-perform available dynamical models in the aspects of reliability of probabilistic forecasts. This area can also be strengthened through application of AI and Machine Learning.

ii. Probabilistic seasonal climate forecasts

The seasonal climate forecast products from Southern African Regional Climate Outlook Forums (SARCOF) process; and those from other Centres that generate seasonal climate forecasts; are mainly presented in probabilities of above-normal, near-normal and below-normal terciles rainfall distributions based on the long term climatology of the regions for which the forecasts are produced. The advantage of showing all three forecast probabilistic distributions together is that all scenarios are described. This provides greater guidance to users that although one particular outcome might be the most likely, other alternative outcomes can also occur. Thus probabilistic forecasts enable better decision making by end users for risk reduction; and to optimize on favourable conditions. Therefore, the event for which the rainfall forecast probabilities are provided must accurately represent the climate sensitivity of the particular user community.

IV. Climate Science for the Benefit of Agricultural Production

Climate services can be defined as decision support tools developed based on a process of transforming climate information into relevant advisory services that assist decision-making by individuals and organizations of a society, Tall, et al (2018). A climate service provides tailored, salient and usable advisories for policy-makers and vulnerable communities, based on available climate information. Most of all, a climate service must respond to user needs (Tall and Njinga; 2013). When connected to crop simulation models, seasonal forecasts can be used to assess implications for crop productivity, as well as to guide crop management decisions (Capa-Morocho, et al, 2016).

Also, tools have been developed to assist in early action using weather and climate forecasts (Lopez, A et al: 2020). Studies have shown that crop yield prediction is benefitting in applying seasonal climate forecast (Coelho and Costa: 2010). Weather generating models can now be adapted for crop capability prediction using probabilistic seasonal climate forecasts, (Han and Ines (2017). These initiatives can be exploited to benefit decision support in agriculture and other areas. This provides options within climate change adaptation strategies.

SARCOF Hit Ratio and False Alarm Rate

The usability of SARCOF products can be assessed through a process of analyzing of Hit and False Alarm Rates. Given a set of forecasts and verification pairs over a period with n discrete event intervals (i.e., months over a series of rainy seasons), they can be summarized in a contingency table as follows:

Table 1 Seasonal climate forecast contingency table

		Observed		
		Yes	No	
Forecast	Yes	a	b	a + b
	No	c	d	c + d
		a + c	b + d	n = a + b + c + d

Table 1 shows four possible outcomes:

An event is forecast and the event occurs (a)

An event is forecast and the event does not occur (b)

An event is not forecast and the event occurs (c)

An event is not forecast and the event does not occur (d).

In this table $n = a + b + c + d$ is the total number of forecast observation pairs (for instance the number of days in the record for daily precipitation forecasts). The sample estimate of the climatological probability of the extreme event is

s = (a + c)/n. The variables a, b, and c can be used to define two key parameters: The Hit Rate and the False Alarm Ratio.

Hit Rate: H = a/(a + c), i.e. H is the sample estimate of the likelihood of event occurring. False Alarm Ratio:

FAR = b/(a + b) i.e. FAR is the sample estimate of the probability of the event not happening given a forecast probability.

Using SARCOF products, time series of Hit Rate and False Alarm Rate was plotted for period 2000-2016, Figure 1. This clearly shows that in the SARCOF seasonal rainfall prediction accuracy improves as years progress. For instance, it can be seen that in 2016, the False Alarm Rate went down to 0.02 while the Hit Rate went up to 0.98. The quality of the inputted SCF into crop-weather models is important for reliability of crop yield prediction.

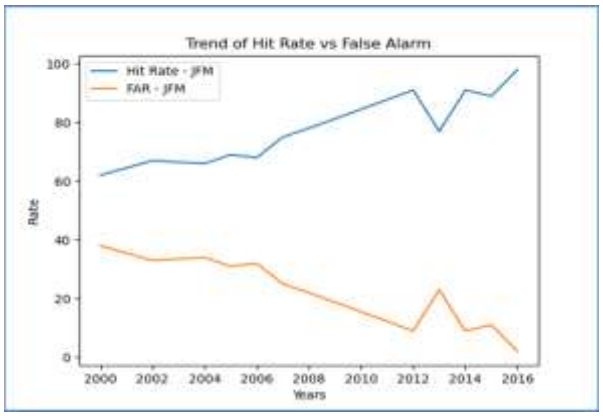


Figure 1. SCF Hit rate (blue line) and False Alarm Rate (brown line) for January-March, 2000-2016.

The above prediction accuracy and reliability trends of SARCOF products can be exploited as trusted input in generating CIS-based Decision Support Tools (DST) in Agriculture for great Socio-Economic Benefit of countries.

V. SEASONAL CLIMATE FORECASTS FOR CROP YIELD MODELLING.

Since 1960, computers allowed synthesizing detailed knowledge on plant physiological processes in order to explain the functioning of crops as a whole, Choudhary (2018). Insights into various processes were expressed using mathematical equations and integrated in crop simulation models. Crop growth simulation models have now become usual tools for agricultural research and production systems.

These are used in combination with seasonal climate forecasts as input for the purposes of generating crop yield projections with long lead times, typically three months in advance. However, crop growth is more complex and occurs continuously in time, thus requiring at least daily inputs of weather parameters, while the seasonal forecast for these parameters are currently only possible in three months blocks.

In order to use SCF as input into crop growth simulation models, there is need to reconfigure SCF outputs as daily weather realizations. Past studies on the behaviour of stochastic weather models have provided a solid foundation for using weather generators to reproduce synthetic daily sequences from seasonal climate forecasts (Wilks 1992, Katz 1996, Mearns et al. 1997). Methods for conditioning weather generator parameters on seasonal predictions or predictors include: estimating parameters from years with a particular categorical predictor value (Katz & Parlange 1993, Grondona et al. 2000, Katz et al. 2003), regressing parameters against a seasonal predictor (Woolhiser et al. 1993), predicting from General Circulation Model (GCM) output fields using multivariate statistical downscaling (Cantelaube & Terres 2005, Feddersen & Andersen 2005, Marletto et al. 2005). Other schemes include sampling past years in proportion to forecast shifts from climatological tercile probabilities to estimate parameters (Wilks 2002).

The crop growth simulation models require daily weather data, soil surface and profile information, and detailed crop management as input. One such model, the Decision Support System for Agrotechnology Transfer (DSSAT) is a software application programme that comprises crop simulation models, developed at the University of Florida, USA, Hoogenboom et al (2019). DSSAT combines crop, soil, and weather data bases with crop models and application programmes to simulate outcomes of crop management strategies at a daily step. Meantime, the development of decision support system based SCF requires access to, and processing of input parameters, for example recordings of seasonal rainfall, maximum and minimum temperatures, sun radiation, longitudes and latitudes for the rainfall homogeneous zones.

All the above parameters are necessary input into a software package called Climate-Agriculture-Modelling and Decision Tool (CAMDT) developed at International Climate for Society (Han and Ines, 2017). CAMDT is a weather generating algorithm, originally developed for running yield projections for rice cultivars grown in the Bicol Region of Philippines. This was adapted for different cultivar varieties in generating crop yield prediction in countries of Sub-Saharan Africa.

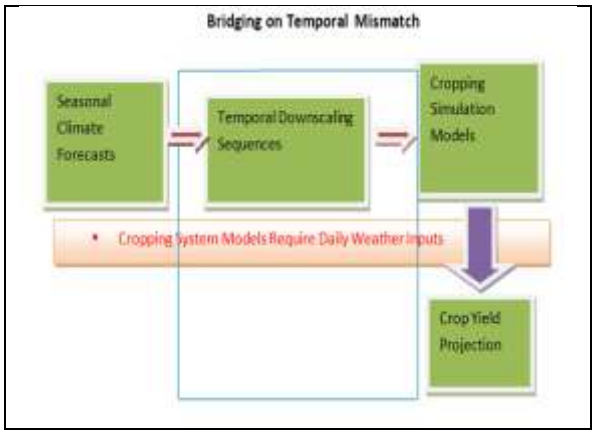


Figure 2 Schematic of temporal downscaling of SCF as input into crop yield models

The CAMDT has a functionality to disaggregate a probabilistic SCF into daily weather realizations (either a parametric or non-parametric disaggregation method) and to run DSSAT-CSM-Rice. It has a convenient graphical user-interface which allows easy implementation of several "what-if" scenarios for non-technical users and visualizes the results of the scenario runs (Ines et al, 2017). The 30-year historical daily weather parameters in solar irradiance, maximum and minimum temperature and precipitation required to run CAMDT were obtained from NASA data sets accessible using the link:
https://power.larc.nasa.gov/data-access-viewer/
A schematic of flow chart on preparation of forecast data for the DSSAT is shown in Figure 2. CAMDT works such that the process begins with a seasonal forecast box in the top left which is downscaled into daily weather realizations in the top middle box which feeds into, and drives, the crop growth simulation model which then lends itself into the crop yield projection at the bottom right. CAMDT, therefore, provides input into DSSAT for the latter to assess the crop cultivar phenology, thereby resolving the challenge between the tercile-based probabilistic three-monthly seasonal forecasts and daily requirements of the climate elements by the crop growth simulations. Once this is provided then the crop growth simulations are possible to generate realistic yield outcomes. This will lead to specific crop capability projections. In turn, this will greatly benefit planning of agricultural production systems. In the study we used crop cultivars with details on coefficients are found in the DSSAT library.

Yield Estimation using Exceedance Curves

Once we establish reliable, skillful SCF with lead times of up to between three and four months these can be inputted into crop weather simulation models for yield predictions. Such crop yield forecasting capability provides considerable planning advantage for the farmers. SCF based crop yield modelling generates outputs such as yield exceedance curves. Exceedance probability (P) is computed as $P = m \times (n + 1)-1$, where m is the rank of a forecasted yield (m=1 is the largest value) and n is the total number of simulated years (yields), Han and Ines (2017). The full distributions of the predicted yields from the daily weather realizations downscaled from the given SCF are shown in Figure 3 (a), (b).

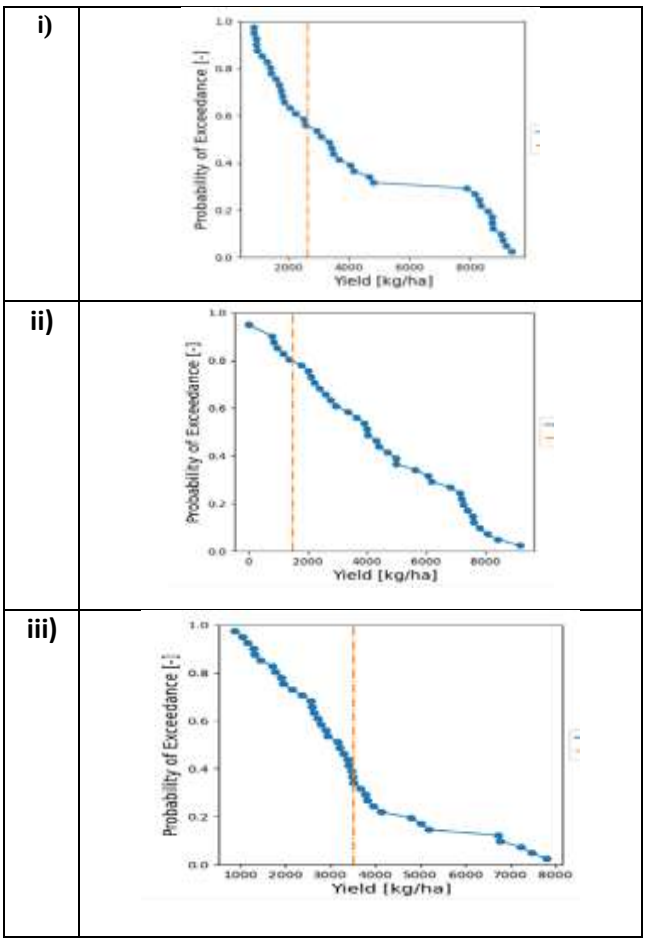


Figure 3 a) Yield exceedance curves based on three-month lead forecasts, with planting date 15 December 2009 (i) rice, (ii) maize and (iii) for Tutume, Botswana for above-normal rainfall. Dotted vertical orange line gives the yield for 2009/10

SCF can be used for deciding what to be preferred planting distributions of different cultivars of rice, maize and sorghum. However, it was also observed that the crop varieties obtained from DSSAT library for the experiments exhibit higher yield performance than those used locally.

Projected yield exceedance curves for above-normal and below-normal rainfall for the three cultivars were generated for Tutume (20.50° S, 27.04° E) in the Central District of Botswana, about 50 km from the Zimbabwe border. The yield curves are shown in Figure 3 (a) and (b). As an experimental study, the

CAMDT model is used in hindcast i.e. provide a target growing season 'forecast' among those in years of the training period in the data series. In this case a randomly chosen particular growing season (2009/10) observed rainfall was forecast. The results for Tutume showed mean yields such that rice would have been about 2500 kg/ha, maize, 1600 kg/ha while sorghum would have been 3500kg/ha. Using SCF three month in advance, projected to result in above-normal rains in Tutume, there would have been nearly 60% chance of obtaining 2500kg/ha for rice which chance would drop to only 30% if below-normal rains were to be expected. For maize yield projection there would have been nearly 80% chance of obtaining 1800kg/ha. To achieve the same tonnage if below-normal rains were expected the probability of exceedance would drop to under 70%. The best yields were possible for sorghum 3500kg/ha.

If above-normal rains were projected for the season, then sorghum would have 40% chance of achieving the target of 3500kg/ha. This probability would decrease to 30% for below-normal rainfall expectation. In other words the best bet for relatively good production in the area was to grow sorghum. So, clearly availing timely skillful SCF in crop growth simulations can provide crop yield exceedance curves which are very important decision support. This type of scenario generation would assist the farmer to decide how much of the farm s/he will put under a particular crop for a particular season based on the SCF provided to the farming community and the policy-makers. There were also results of different stations across other SSA countries over different years. It must also be pointed out that there is need for carrying out field experiments with various crop cultivars in order to calibrate the yield against environmental and management options as provided by model outputs.

.

Meantime, there is need for caution in the interpretation of the actual yield quantities generated by the model which appear to be more on the high side. Suffice to say the qualitative yield exceedance curves are quite instructive as they vary proportionately in accordance with the respective forecast tercile categories for each of the crop cultivars. As an interim measure, these differences can be partly resolved through application of AI.

The development of crop yield models should integrate research in order to identify advances and gaps in knowledge. Such outcomes have many benefits for applied research, teaching, and resource management, policy analysis and crop production forecast at national and regional level.

CAMDT enables the stakeholders to have an overview of the feasibility of a desired prediction horizon (farming season) and offer scientifically backed advice to the farmers of the particular homogeneous rainfall zones.

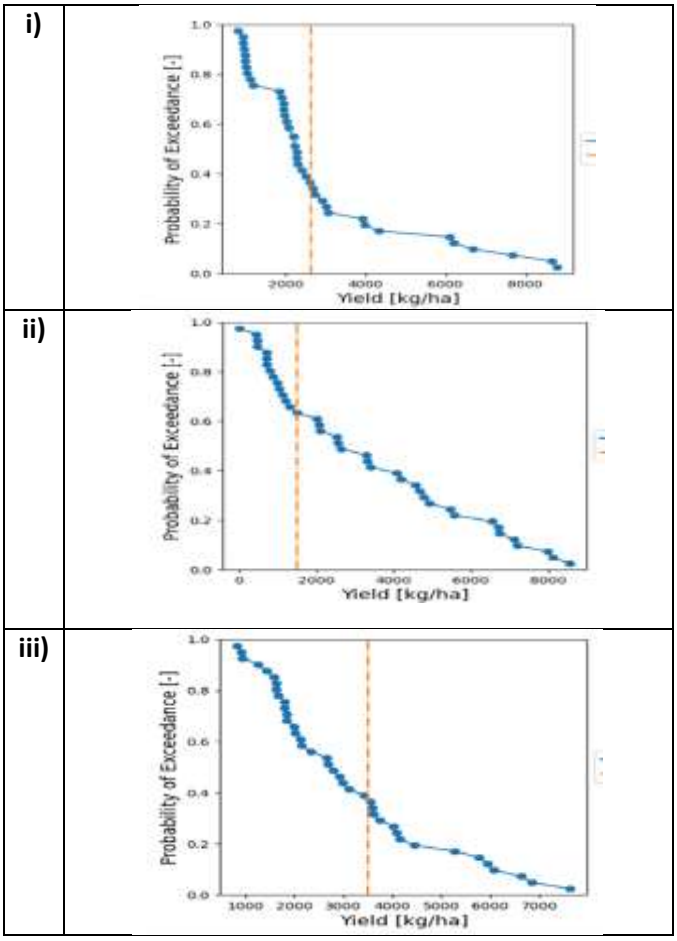


Figure 3 b) Yield exceedance curves based on three-month lead forecasts, with planting date 15 December 2009 (i) rice, (ii) maize and (iii) for Tutume, Botswana for below-normal rainfall. Dotted vertical orange line gives the yield for 2009/10

VI. CONCLUDING REMARKS

Variability in the climate system often has enormous impact on agricultural productivity: when the prevailing climatic patterns are favourable they can enhance production; but when they are adverse they often lead to food insecurity. It is, therefore, critical to find ways of alerting the farmer and policy makers well ahead of planting horizon what yields will likely to be. The current study is an attempt to achieve that. The study showed how CAMDT an algorithm for temporally downscaling seasonal climate forecast into synthetic daily weather generating is used to drive DSSAT crop growth simulation models. Using crops such as rice, maize and sorghum in countries in Sub-Saharan Africa, it was demonstrated that crop yield prediction

premised on long-lead SCF can be an important decision support tool. The tool can enable a farmer to minimize her/his losses in droughts and take advantage of favourable seasons to maximize margins. CAMDT as a wrapper of DSSAT is extremely important as decision support to both farmers and policy makers for making strategic and tactical plans for more optimal agricultural production systems and help ensure food security and increasing margins to the former. This tool can also assist policy makers in their plans for grain procurements when there is deficit in order to ensure food security; or for exports of surplus grain well ahead of harvests. Such is the promise of seasonal climate forecasts driven crop yield models. Climate science will advance even more rapidly thanks to ability for humankind to exploit capabilities of Artificial Intelligence and Machine Learning. In order to register greater progress more quickly, arrangements must be always in place from the climate scientists and practitioners on one hand and wider user community on the other hand to interact to share experiences in order to help farmers become more efficient. This will help accelerate the processes required to make agricultural production systems to be more efficient, thereby contributing to food security in Sub-Saharan Africa.

REFERENCES:

Cantelaube, P. and J. M. Terres (2005), Seasonal weather forecasts for crop yield modelling in Europe. Tellus **57A**:476-487.

Capa-Morocho M, B Rodríguez-Fonseca, M Ruiz-Ramos, (2016), Agricultural and Forest Meteorology **226–227**, 213-228, https://doi.org/10.1016/j.agrformet.2016.06.007.

Carbone G. J. (1993), Considerations of meteorological time series in estimating regional-scale crop yield. J Clim **6**:1607–1615.

Choudhary D (2018), Crop Growth Simulation Model for Agriculture. Submitted to Dept. of Agricultural Meteorology, B.A. College Of Agriculture Anand Agricultural University 338110. DOI: 10.13140/RG.2.2.35505.28008.

Coelho, C A S and S M S Costa (2010), Challenges for integrating seasonal climate forecasts in user applications. Current Opinion in Environmental Sustainability **2**:1–9. DOI 10.1016/j.cosust.2010.09.002;
https://www.cptec.inpe.br/pesquisadores/caio.coelho/Coelho_and_Costa_2010.

Drosdowsky W., and L. E. Chambers (2001), Near-global sea surface temperature anomalies as predictors of Australian seasonal rainfall. J. Climate **14** (7): 1677–1687. https://doi.org/10.1175/1520-0442(2001)014<1677:NACNGS>2.0.CO;2.

Grondona M.O., G.P. Podestá, M. Bidegain, M. Marino and H. Hordu (2000), A stochastic precipitation generator conditioned on ENSO phase: a case study in southeastern South America. J Clim **13**:2973 –2986.

Hallegatte S. (2012), A cost effective solution to reduce disaster losses in developing countries: hydro-meteorological services, early warning and evacuation. World Bank Policy Research Paper No. 6058. The World Bank, Washington, D.C. USA.

Han, E- J. and A. V. M. Ines (2017), Downscaling probabilistic seasonal climate forecasts for decision support in agriculture: A comparison of parametric and non-parametric approach. Climate Risk Management **18**, 2017, Pages 51-65. https://doi.org/10.1016/j.crm.2017.09.003 FEWSNET.

Hoogenboom, G., C. H. Porter, V. Shelia, K. J. Boote, U. Singh, J.W., White, L.A. Hunt, R. Ogoshi, J. I. Lizaso, J. Koo, S. Asseng, A. Singels, L. P. Moreno and J. W. Jones (2019), Decision Support System for Agrotechnology Transfer (DSSAT) Version 4.7 (www.DSSAT.net). DSSAT Foundation, Gainesville, Florida, USA.

Jones, J.W., G. Hoogenboom, C.H. Porter, K. J. Boote, W.D. Batchelor, L. A. Hunt, P. W. Wilkens, U. Singh, A. J. Gijsman, and J. T. Ritchie (2003), DSSAT Cropping System Model. European Journal of Agronomy **18**:235-265.

Katz, R.W. M. B. Parlange, (1993), Effects of an index of atmospheric circulation on stochastic properties of precipitation. Water Resour Res **29**:2335 –2344.

Katz R. W., M. B. Parlange and C. Tebaldi (2003), Stochastic modeling of the effects of large-scale circulation on daily weather in the southeastern US. Clim Change **60**:189 –216.

Lopez, A., Coughlan de Perez, E., Bazo, J., Suarez, P., van den Hurk, B., & van Aalst, M. (Accepted/In press). Bridging forecast verification and humanitarian decisions: A valuation approach for setting up action-oriented early warnings. Weather and climate extremes. https://doi.org/10.1016/j.wace.2018.03.006.

Marletto V., F. Zinoni, L. Criscuolo, G. Fontana, S. Marchesi, A. Morgillo, M. Van Soetendael, E. Ceotto and U. Andersen (2005), Evaluation of downscaled DEMETER multi-model ensemble seasonal hindcasts in a northern Italy location by means of a model of wheat growth and soil water balance. Tellus **57A**:488 – 497.

Mearns L. O., C. Rosenzweig C and R. Goldberg (1997), Mean and variance change in climate scenarios: methods, agricultural applications, and measures of uncertainty. Clim Change **37**: 367– 396.

SADC (September, 2016), Regional Humanitarian Appeal, Addendum.
https://www.sadc.int/files/2014/7463/4292/Appeal_Document_ADDENDUM_20160923.pdf.

Tall A, J. Y. Coulibaly and M. Diop (2018), Do climate services make a difference? A review of evaluation methodologies and practices to assess the value of climate information services for farmers: Implications for Africa. Climate Services **11**, 1-12;https://doi.org/10.1016/j.cliser.2018.06.001).

Tall, A. and J. Njinga, (2013), Developing a methodology to evaluate climate services for farmers in Africa and South Asia workshop report. Copenhagen, Denmark: CGIAR Program on Climate Change, Agriculture and Food Security. Retrieved from: www.ccafs.cgiar.org. Google Scholar).

UNECA (2020), Socio-Economic Benefits for Climate Information Service as applied to Disaster Risk Reduction. In press.

Wilks D. S. (1992), Adapting stochastic weather generation algorithms for climate change studies. Clim Change 22:67– 84

Wilks, D. S. (2002), Realizations of daily weather in forecast seasonal climate. J Hydrometeorol **3**: 195 –207.

Woolhiser D. A., T.O. Keefer and K. T. Redmond (1993) Southern oscillation effects on daily precipitation in the southwestern United States. Water Resour Res **29**:1287–1296.

Zhang, X. C. (2003), Assessing seasonal climate impact on water resources and crop production using CLIGEN and WEPP models. Trans Am Soc Agric Eng **46**: 685 – 693.

World Bank (2019) Zimbabwe: Agriculture Sector Disaster Risk Assessment.
https://www.gfdrr.org/sites/default/files/publication/Zimbabwe%20Agriculture%20Sector%20Disaster%20Risk%20.

Patrones de Circulación Asociados a la Precipitación en la Región de Transición de la Amazonía-Andes

Melissa Medina
Departamentos de Ciencias
Universidad Privada de Ciencias Aplicadas
Lima, Perú
Programa de Doctorado en Recursos Hídricos
Universidad Nacional Agraria La Molina
Lima, Perú
Subdirección de Ciencias de la Atmósfera e Hidrósfera
Instituto Geofísico del Perú
Lima, Perú
email:20200864@lamolina.edu.pe

Eusebio Cisneros
Departamento de Física y Meteorología
Universidad Nacional Agraria La Molina
Lima, Perú
email: ecisneros@lamolina.edu.pe

Arturo Sanchez
Programa de Pós-Graduação em Computação Aplicada (CAP)
Instituto Nacional de Pesquisas Espaciais (INPE)
São Paulo, Brasil.
email: arturo66cta@gmail.com.pe

Abstract

The influence of rainfall in an area such as the Bolivian Peruvian Altiplano is of great importance for economic activities such as agriculture. That is why the study focused on identifying wind patterns at 200 hPa that are related to precipitation in the summer and winter seasons, with emphasis on extreme events associated with rains in the region of the Bolivian Peruvian highlands and part of the Bolivian Amazon base.

For the classification of the CPs, the hybrid method of data grouping based on neural networks, Self-Organizing Maps (SOM) and Hierarchical Ascending Classification (CAH) was used, reanalysis data at 200 hPa of ERA - Interim with a resolution of 0.75º × 0.75º for the period from 1979 to 2016 in the area of interest (15º S - 25º S, 80º W - 60º W) and each pattern was associated with the OLR (Outgoing Longwave Radiation) data obtained from the NCAR / NOAA website with a resolution of 2.5 ° x 2.5 ° from 1979 to 2016 and the possible convective systems associated with characteristic wind behaviors were listed. The patterns identified, for summer were 6 and for winter they were 7, they showed defined time cycles with durations of between 3 and 11 days, associated with extreme events such as floods and droughts related to the activity of the monsoon system in summer and the propagation of extratropical waves. in winter. The results are consistent with the behavior of precipitation in the meteorological stations in the study area provided by SENAMHI-Bolivia and SENAMHI-Peru and OLR. These results showed new perspectives on extreme hydrological events associated with atmospheric circulation in certain circulation patterns.

Keywords—extreme events, atmospheric circulation patterns, Andes, Amazon, neuronal networks.

I. INTRODUCCIÓN

La cuenca del Amazonas tiene un importante rol en el sistema climático global y es el sistema hidrológico más grande del mundo, el 11% de su territorio corresponde a la Cordillera de los Andes donde la región de transición entre las tierras bajas y el terreno montañoso es considerada la más lluviosa de toda la cuenca amazónica [8]. La Amazonía está ubicada entre los 5∘N-20∘S, por lo que tiene un clima tropical con regímenes de precipitación que siguen un patrón estacional y difieren de una región a otra [13]. En el sur, existe un régimen tropical austral con una precipitación máxima en diciembre-enero-febrero (verano austral) relacionada con el sistema monzónico de América del Sur ([5], [23]) que trae vientos húmedos del norte de las regiones ecuatoriales, especialmente del Océano Atlántico. Otra característica importante es la Zona de Convergencia Sudamericana (SACZ), ([5] , [6]) que también contribuye con las precipitaciones en el sur de la Amazonia durante el verano austral.

El Altiplano se caracteriza por tener un clima semiárido y una precipitación media anual entre los 600 mm en el noreste y 200 mm en el sur. Durante el verano, entre los meses de diciembre, enero y febrero ocurre la mayor parte de la precipitación (70%) cuando los vientos del este de altos niveles permiten el ascenso de aire húmedo que alimenta sistemas convectivos en esta área y presentando episodios húmedos y secos con duraciones de entre 5 y 15 días ([9], [13], [17], [18]).

En verano, la precipitación tiene una significativa variabilidad sinóptica, que es explicada en gran medida por la capa límite local de humedad y el viento zonal en el Altiplano [18] y la formación de sistemas convectivos asociados a severas tormentas de corta duración ([28],[29]), que liberan calor latente mientras que los flujos de viento en superficie se desplazan hacia el este, produciento un mecanismo de retroalimentación positiva. En invierno, los episodios de incursiones de vientos del sur han sido estudiados en diferentes áreas alrededor del mundo, tales como el comportamiento del sistema del Monzón de Asia y sus características [24]. En Sudamérica, los vientos en bajos niveles, se desplaza en dirección de sur a norte en el continente y genera cambios drásticos en el sistema térmico y dinámico de la atmósfera a latitudes medias y alta [26], [27]. Estos mecanismos, influyen más en el invierno austral ([19],[29]), los *Cut off Low* o Bajas Segregadas son uno de los procesos que influye en las precipitaciones, ocurren en la alta y media tropósfera debido a la contribución de aire frío del jet streak. Estos sistemas fomentan el incremento en el gradiente zonal de presión que favorece el flujo de vientos del norte proveniente de la Amazonía [11] e intensifica los vientos hacia el norte entre Argentina, Bolivia y el sureste de Brasil a causa del viento polar [26]. En América del Sur, en la referencia [10] realizó estudios sobre patrones relacionados con la

XXX-X-XXXX-XXXX-X/XX/$XX.00 ©20XX IEEE

circulación de vientos a bajos niveles y su influencia en la precipitación, así como en la referencia [8], donde los puntos o zonas más calientes denominadas hotspots, que son zonas en Perú y Bolivia que durante todo el año son húmedos por influencia de la circulación atmosférica influenciada por la presencia de la cordillera de los Andes y las principales características climáticas de los Andes, con énfasis en la media circulación atmosférica a gran escala, las interconexiones hidroclimáticas Andes-Amazonas y los ciclos de precipitación diurnos y anuales más característicos [8]. En clima, estas metodologías se han usado para clasificar los regímenes de tempestades en el norte de México y Texas [7], así como en Sudamérica [10].

Por lo tanto, la zona de transición entre los Andes y la Cuenca Amazónica es de suma importancia para entender cómo se caracteriza la distribución espacial y temporal de la precipitación en las estaciones de verano e invierno y así analizar los patrones de circulación asociados a eventos extremos secos o húmedos. Este trabajo recopila un conjunto de datos diarios de lluvia observada en Perú y Bolivia, así como datos de reanálisis para diferentes variables meteorológicas como vientos y presión a diferentes niveles.

II. MATERIALES Y MÉTODOS

A. Descripción del área de estudio

El área del altiplano peruano boliviano es una planicie extensa con altitudes medias entre los 1000 msnm hasta los 6000 msnm, esto debido a la presencia de la cordillera de los Andes, causando influencia, no sólo en el territorio, sino también en el clima. El factor topográfico y la presencia de glaciares tropicales, hace que se presente un clima extremo que limitan la vida, flora y fauna, en condiciones extremas [16].

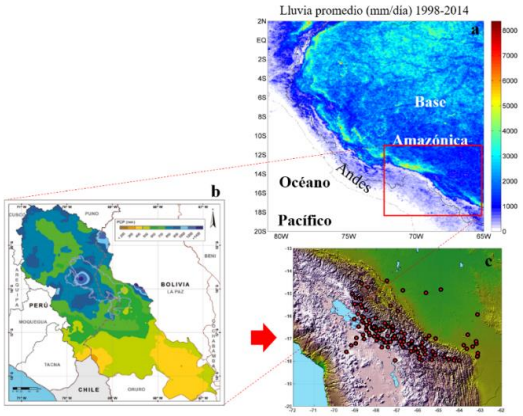
B. Datos

Para definir los Patrones de Circulación (CP) se usó datos diarios de vientos a 200 hPa de reanálisis de ERA-Interim del European Centre for Medium-Range Weather Forecasts (ECMWF) e información diaria interpolada de *Outgoing Longwave Radiation* (OLR) de la website de la NCAR/NOAA para la región de 15º S – 25º S y 80º W – 60º W para un periodo de 1979 – 2016, así como altura geopotencial. La circulación atmosférica a 200 hPa tiene gran influencia tanto en verano, debido a la presencia del alta de Bolivia que se intensifica en el continente posicionada cerca a la región de Santa Cruz [25], mientras que en invierno, las bajas segregadas y friajes cumplen un papel importante debido a la circulación asociada en la tropósfera [24]. Adicional a ello, se usó datos históricos de precipitación observada de 173 estaciones meteorológicas de la región de estudio (15º S – 25º S y 80º W – 60º W) y alturas variables entre 156 m hasta 4 460 m en el periodo de 1979 al 2016 proporcionadas por Servicio Nacional de Meteorología e Hidrología-SENAMHI del Perú y Bolivia. (Figura 1)

C. Metodología

Para identificar los patrones de circulación de vientos a 200 hPa se utiliza el método de SOM y Clasificación Ascendente Jerárquica descrito en [10] para ello se analiza las estaciones de verano, los meses de diciembre, enero y febrero (DEF), e invierno, los meses de junio, julio y agosto (JJA) y los datos de vientos (componentes zonal y meridional). Así mismo, estos datos fueron estandarizados para cada punto de grilla del área de 15º S – 25º S y 80º W – 60º W. Otras variables son descritas en la Tabla 1.

El algoritmo de SOM o es una técnica de clasificación estadística del tipo no lineal que está basado en el



Precipitación media annual (mm/año, 1998-2014) estimada usando datos de TRMM PR 2A25. (b) Precipitación media anual para la Cuenca del Titicaca (mm/año, 1981-2010) Fuente: Senamhi (2017). (c) Ubicación de estaciones meteorológicas.

Figura 1: (a) agrupamiento no supervisado ([20], [21]). Este algoritmo realiza una proyección no lineal de los datos de entrada en una matriz de salida bidimensional regular donde cada componente se denomina neurona. Cada neurona queda definida por su posición en la matriz o mapa y un vector referente.

La forma y el tamaño de la matriz de salida condicionan el número de relaciones entre neuronas, esta relación y conexión con las neuronas vecinas se denomina vecindad, este criterio se basa en la mínima distancia. Cada neurona de la capa de salida está completamente conectada a la capa de entrada, la elección de las dimensiones del mapa depende en gran medida de lo que desee modelar y del nivel de detalle que requiere el estudio. Por ejemplo, en este estudio probamos mapas de neuronas 5×5, 7×7 y 10×10. Para ello, al trabajar con estaciones del año, la mejor alternativa fue trabajar con mapas de 7×7 basado en la metodología usada en [10]. Entonces, la capa de salida consta de 49 neuronas con cartas de dimensiones 7×7, luego cada neurona de salida queda definida en la carta de Kohonen por el vector referente.

Se obtiene una grilla de 285×108 puntos, cada punto de grilla de latitud y longitud con una serie de tiempo de 3340 días para DEF y 3496 en JJA y dos variables correspondientes a las componentes del viento *u* y *v*. Entonces, las neuronas vecinas tienen vectores referentes parecidos y, por lo tanto, son también similares en el espacio. Por lo tanto, cada vector referente es un vector de 4 dimensiones, donde la localización geográfica está definida por los datos de entrada y cada vector representa una situación meteorológica.

El segundo análisis usado fue la Clasificación Ascendente Jerárquica, esta vez sobre el espacio reducido que representa el mapa de neuronas, para formar grupos de neuronas o clases con características similares, de esta manera, cada clase agrupa condiciones atmosféricas similares y cada una puede ser considerada como un tipo de clima característico.

Este método consta de tres pasos, primero se determina las distancias entre cada par de vectores referentes basado en el criterio de distancias Euclidianas. Segundo, agrupar iterativamente las clases en un árbol jerárquico binario basado en el criterio de mínima variabilidad entre los grupos y tomando la información anteriormente generada por la distancia entre los vectores, se agrupan los pares más cercanos hasta formar grupos de mayor tamaño y característicos en un dendograma. Si bien es cierto, este método reduce la información diaria, no reduce las características

Tabla 1:Características de los Datos de Reanálisis ERA – Interim (ECMWF) y OLR (NOAA), para la región de 60º S – 60º N y 150º O – 150º E.

Variable	Resolución Espacial	Resolución Temporal	Nivel Vertical	Periodo
Viento zonal y meridional ("*u*" y "*v*")	0,75° x 0,75°	Datos de 0 h, 6 h, 12 h y 18 h diarios	850 hPa, 500 hPa y 200 hPa	1979 – 2016
Altura geopotencial	0,75° x 0,75°	Datos de 0 h, 6 h, 12 h y 18 h diarios	850 hPa, 500 hPa y 200 hPa	1979 – 2016
Radiación de Onda Larga (OLR)	2,5° x 2,5°	diaria		1979 – 2016

meteorológicas predominantes. Por último, se generó una lista en la cual a cada día del periodo de estudio para DEF y JJA se le asigna el número correspondiente a uno de los patrones de circulación. De manera que cada patrón de circulación (CP) representa un conjunto de días con circulación atmosférica semejante. La metodología utilizada se muestra en la Figura 2.

D. Análisis Temporal de los CPs.

Se analizó las características temporales de cada *CP*, para ellos se determinó probabilidad que existe que un patrón de circulación pase a otro patrón en una cierta cantidad de días posteriores denominada transición:

$$CPm(t)=CPn(t+n)$$

Siendo t el día a analizar que pertenece a la ClassData o grupo de días de cada estación del año *n=1,2,3,… días posteriores* y la probabilidad que un patrón de circulación permanezca los días siguientes al primer día que este se presenta, conocida como persistencia:

$$CPm(t)=CPm(t+i)$$

donde t es el día a analizar que pertenece a la ClassData o grupo de días de cada estación del año y *m=1,2,3,… patrones de circulación correspondientes al día t.* Considerando que *i=1,2,3,… es el número de días posteriores al día t.* Luego, analizó la influencia que presenta en las variables meteorológicas involucradas realizando un análisis de la estructura vertical con la información de altura geopotencial estandarizada a diferentes niveles de presión: 850 hPa y 200 hPa, para entender la dinámica en la baja y alta atmósfera y para cada CP. Para analizar la precipitación, se seleccionó los días correspondientes a cada patrón característico para los datos de anomalía estandarizada de OLR y precipitaciones de las estaciones meteorológicas y se promedió los días agrupados asociadas a los CPs.

III. RESULTADOS

Para DEF, los escenarios predominantes cumplieron con un ciclo temporal definido con una duración entre 3 a 8 días (Figura 3) cuyo comportamiento estacional característico queda definido con 6 patrones de circulación, estos esquemas temporales quedan definidos por las transiciones y persistencias para el día siguiente de ocurrido el CP.

El CP5 representa el 25.5% del total de días del DEF y se determinó la tercera probabilidad de ocurrencia significativa al 90% mayor o igual a 11%. Se define como el patrón inicial al patrón CP1 por presentar la mayor persistencia entre los CPs (54%) y 18% de pasar al día siguiente de ocurrido este comportamiento al CP4, el CP4 muestra un 51% de persistencia y un 11% de pasar al CP2, el CP2 tiene un 51 % de continuar un día más o de pasar al CP5 con una transición de 26 %, una vez en el CP5 la persistencia es del 72% y la transición es del 17 % para el CP6, mientras el CP6 muestra un 65% de mantener el comportamiento un día más y de pasar al CP3 un 8% y retornar al CP4 es de 12%, ya en el patrón CP3 la persistencia es de 54% y de transición del 19% al CP4.

Una de las alternativas al comportamiento cíclico de verano es el ciclo con la tercera probabilidad más alta que puede ir al CP1, CP2, CP5, CP4 y nuevamente al CP1.

En JJA la probabilidad de ocurrencia con un día posterior y la tercera probabilidad mayor o igual al 12% significativa al 90 por ciento, se enumeró los CPs del 1 al 7. Estos escenarios mostraron un ciclo temporal definido con una duración de entre 3 a 11 días como máximo. En este caso, se consideró como patrón CP7 el primer patrón obtenido. (Figura 4).

El ciclo empieza con el patrón CP7 que presenta un 58% de persistir un día más, mientras que tiene un 22% de pasar al CP3, el CP3 muestra un 38% de persistencia y un 36%de pasar al CP4, el CP4 tiene un 37% de continuar un día más o de pasar al CP5 con una transición del 31%, una vez en el CP5 la persistencia es del 42% y la transición es del 39% para el CP6, mientras el CP6 muestra un 54% de mantener el comportamiento un día más y de pasar al CP1 un 22%, ya en el patrón CP1 la persistencia es de 41% y de pasar al CP7 del 28%, siendo la carta de JJA la que presenta un comportamiento bien definido comparado. La carta que no muestra un comportamiento vinculado dentro del ciclo es el CP2, que usualmente se presenta en casi todas las cartas como la tercera probabilidad posible, pero solo en dos de ellas es significativa. Ello puede estar relacionado a un patrón recurrente o característico de esta estación.

IV. DISCUSIÓN

En verano se observa actividad convectiva en gran parte de la cuenca amazónica y la cuenca de La Plata, los vientos provenientes del oeste se van intensificando y moviliza el sistema convectivo la parte norte del Perú y noreste de Brasil (CP2), mientras que un sistema de baja presión en altos niveles se posiciona en Argentina, estos sistemas se intensifican a lo largo del lado occidental de la cordillera de los Andes y la base amazónica. La confluencia de vientos en CP3 se origina por la posición de los sistemas de alta y baja presión en el continente, incrementándose la velocidad de los vientos sobre el altiplano boliviano, generando precipitaciones en CP4 (colores fríos o azules), estos patrones muestran la presencia de la Alta de Bolivia que favorece a las precipitaciones en el Altiplano y parte de la Amazonía Boliviana. Progresivamente estas condiciones van cambiando sobre el continente (CP5).

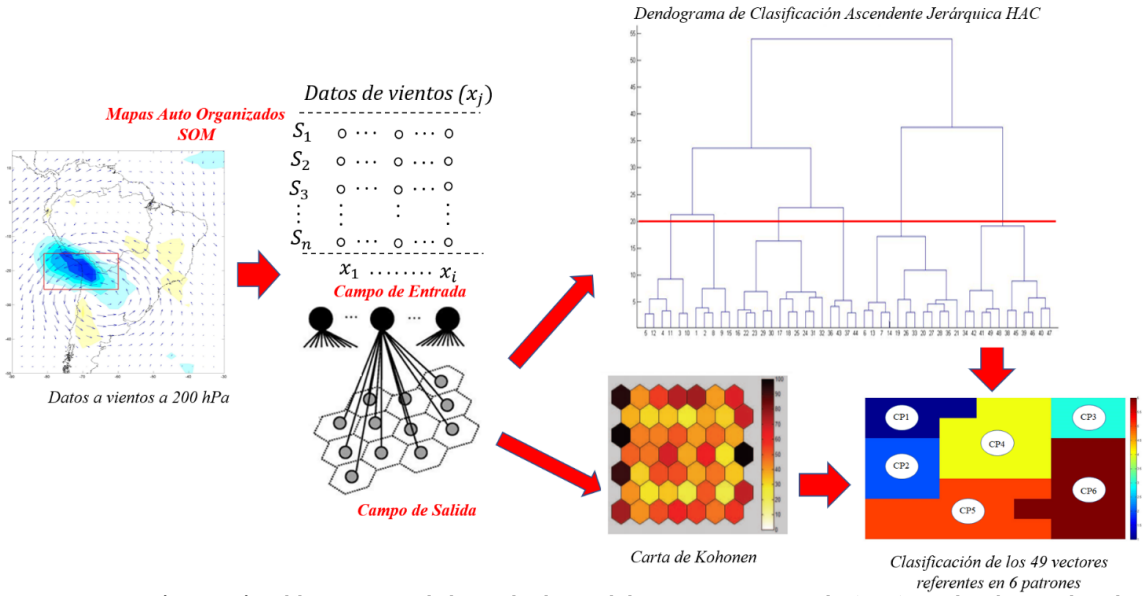


Figura 2: Representación esquemática del procesamiento de datos realizados usando los Mapas Auto Organizados (SOM) carta de Kohonen y el Dendograma de la Clasificación Ascendente Jerárquica (CAH), se obtiene como resultado la clasificación y determinación del número de CP basado en 49 vectores referentes para verano (DEF). La misma metodología fue empleada en invierno (JJA).

Los vientos se debilitan teniendo días despejados (valores positivos de OLR) y el CP6 muestra vientos del este sobre el continente y un sistema de baja presión en el altiplano a 200 hPa. La convergencia y divergencia de vientos, régimen del norte y régimen del sur de acuerdo con [29], facilitan el transporte de masas de aire de latitudes subtropicales a latitudes tropicales relacionado a la actividad vertical y proceso de mezcla entre altos y bajos niveles atmosféricos, sumado a la barrera orográfica natural cordillera de los Andes ([18], [29]) intensifica progresivamente el comportamiento de los vientos. El CP1, muestra una mayor influencia en las estaciones en la base amazónica, esta precipitación disminuye progresivamente (ver CP2). El CP3 caracteriza la circulación atmosférica que favorece las precipitaciones en esta estación del año. Estas condiciones continúan en el CP4. En el CP5 se observa la disminución de la precipitación en toda el área, mientras el CP6, disminuye aún más en los Andes, esta ausencia de precipitación es notoria comparada con la precipitación en la Amazonía, mostrando un sistema dual característico debido a la presencia del Alta de Bolivia que se comprueba en CP1 y CP4, como lo mencionan ([8], [12]). El CP2 y CP5 muestran un comportamiento asociado a la circulación de la SACZ o Zona de Convergencia del Atlántico Sur, debido a la convergencia de vientos en el este de la cuenca amazónica y la ausencia de precipitación es el Altiplano debido a las anomalías de vientos del oeste, lo cual es típico de una configuración de la SACZ intenso, de acuerdo con la referencia a [3], [4] y [5].

Los sistemas mostrados cumplen con las condiciones características descritas en [10], [25], [27] y [28] relacionadas a la circulación de bajos niveles y la presencia de vientos del noreste como consecuencia de la influencia de los vientos alisios desviados hacia el sureste por la presencia de la cordillera de los Andes y la convergencia de vientos provenientes del Atlántico subtropical para finalmente transportar masas de aire hacia Brasil y Argentina, estas condiciones son mencionadas por [29] para la descripción de SALLJ, la fase madura del SAMS y la presencia del Alta de Bolivia en altos niveles ([9], [13]).

En invierno (Figura 4), el CP1 muestra un núcleo convectivo asociado a valores positivos de OLR (colores cálidos) y vientos del sureste, este tipo de circulación propicia el desplazamiento de flujos de humedad hacia continente ingresando por la costa y posicionándose en el Altiplano y llano amazónico (ver CP2 y CP3). El CP4 y CP5, muestran un sistema de baja presión en altos niveles de la atmósfera y cómo el núcleo convectivo se debilita y propaga hacia la cuenca de La Plata, estos vientos con circulación ciclónica progresivamente van cambiando sobre el continente y propician el ingreso de flujos de vientos secos que se intensifican (CP6 y CP7) y días despejados. Estos vientos se intensifican en todo el altiplano. La precipitación se incrementa durante la formación de las bajas segregadas ([1], [14], [20]).

Estos sistemas favorecen el incremento en la gradiente zonal de presión y propicia el desplazamiento del flujo del norte proveniente de la Amazonía ([15], [28]) e intensifica los vientos hacia el norte, entre Argentina, Bolivia y sureste de Brasil a causa del viento polar y reflejado en los vientos a 850 hPa ([26], [27]). El patrón CP7 está vinculado la una onda de Rossby que se inestabiliza formando remolinos aislados [2], este bloque de aire frío queda segregado. Simultáneamente aparecen anticiclones o sistemas de alta presión que quedan bloqueadas y asociadas a aire caliente. En síntesis, la rotura de la contribuyen al transporte de calor entre el ecuador terrestre y los polos. circulación de los vientos del oeste ocasiona cambios bruscos de tiempo en las respectivas latitudes (frío en las zonas subtropicales y calor en las áreas polares) [17]. Los sistemas convectivos asociados a los CPs muestran el posicionamiento de un sistema de baja presión en continente, mientras que los sistemas de alta presión en los Andes y llano amazónico están asociados a días secos (colores cálidos), intensificándose progresivamente, marcando patrones que muestran la dinámica y circulación atmosférica en el JJA que es regida por la propagación de trenes de ondas desplazándose a 200 hPa y muestran patrones de sequías y precipitaciones intensas [18]. El CP1 agrupa días con déficit de precipitación que se incrementan progresivamente (CP2 y CP3). El CP4 y CP5 muestran ausencia de precipitación en los Andes e incremento de precipitaciones en la Amazonía mostrando un

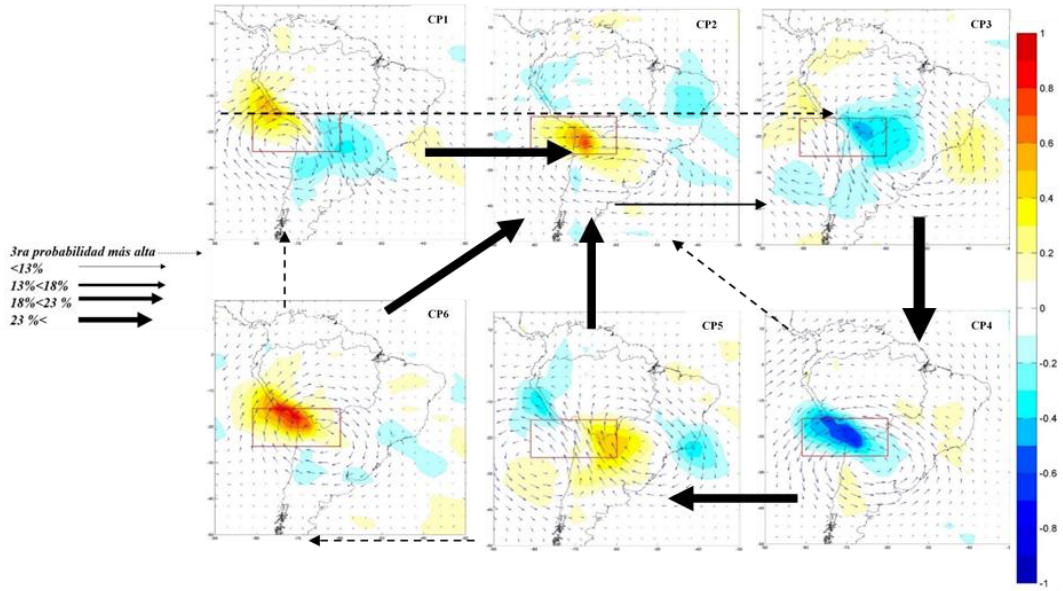


Figura 3: Composición de los patrones de circulación (CPs) denominada carta de circulación para el verano (DEF) de anomalías estandarizadas de vientos a 200 hPa (vectores) y OLR (colores sólidos) para cada punto de grilla y con perido común de 1979 – 2013. Posibles sistemas convectivos presentes (ausentes) en cada patrón están asociados a valores negativos (positivos) de OLR en color azul (rojo). Las fechas indican la probabilidad significativa de transición de un patrón a otro según la prueba t (ρ<0,1). El recuadro de color rojo representa el área donde han sido determinados los CPs.

sistema dual, ocasionado por confluencia de vientos del noroeste en bajos niveles y un sistema de baja y alta presión y acoplados en el eje vertical (celda de Hadley) y la propagación de ondas atmosféricas a 200 hPa [27], mientras que los CP6 y CP7 muestran la ausencia de precipitaciones en casi toda la zona, esto está relacionado al sistema de altas presiones posicionado en el continente que se van intensificando y debilitando en un tren de ondas que se disipa en el continente debido a la cordillera de los Andes, dirigiéndose hacia el ecuador [22].

Conclusiones

Para la estación de verano se identificaron 6 patrones de circulación atmosférica asociados a vientos provenientes del oeste que se van intensificando, desplazando el núcleo convectivo al noreste de Brasil y parte del norte del Perú desde gran parte de la cuenca Amazónica y la cuenca de La Plata.

En invierno se identificaron 7 patrones de circulación atmosférica relacionados a la convergencia de vientos en el altiplano en bajos niveles provenientes del noreste (cuenca Amazónica) y noroeste (Océano Pacífico) propicia el desplazamiento de flujos de humedad hacia el continente, el CP1 muestra días con precipitaciones sobre el valor promedio en el llano amazónico y en estaciones ubicadas en los Andes hay un déficit de precipitación. El CP4 muestra un comportamiento inverso al del CP1, debido a la influencia de los vientos y el Alta de Bolivia sobre el continente, que concuerda con la presencia de la SACZ que se intensifica o debilita. Para JJA se considera como patrón húmedo al CP3 para las estaciones en la Amazonía y como patrón seco al CP6. El comportamiento dual característico entre los Andes y la Amazonía se muestra en las cartas CP2, CP3 y CP4 influenciado por la diferencia altitudinal y confluencia de vientos por los Andes que actúa como barrera natural para el flujo de vientos y transporte de humedad. A 200 hPa, las ondas atmosféricas que se propagan son intensas, formando un sistema de baja presión sobre el área de interés para los CPs.

Agradecimientos

Un especial agradecimiento a Innovate Perú, PNICP – Peru "Contrato 397-PNICP-PIAP-2014" como parte de la iniciativa en investigación del estado peruano que ha proporcionado los fondos para el desarrollo de la investigación y a los docentes de la UNALM e investigadores del IGP como el Dr. Jhan Carlo Esponiza que enriquecieron este trabajo con sus contribuciones y comentarios.

Referencias

[1] Bischoff, S. A., and Vargas, W. M. (2003). The 500 and 1000 hPa weather type circulations and their relationship with some extreme climatic conditions over southern South America. International Journal of Climatology, 23(5), 541–556. https://doi.org/10.1002/joc.894

[2] Campetella, C. M., and Possia, N. E. (2007). Upper-level cut-off lows in southern South America. Meteorology and Atmospheric Physics, 96(1–2), 181–191. https://doi.org/10.1007/s00703-006-0227-2

[3] Carvalho et al. - 2005—Opposite Phases of the Antarctic Oscillation and R.pdf. (n.d.).

[4] Carvalho, L. M. V., Jones, C., and Ambrizzi, T. (2005). Opposite Phases of the Antarctic Oscillation and Relationships with Intraseasonal to Interannual Activity in the Tropics during the Austral Summer. Journal of Climate, 18(5), 702–718. https://doi.org/10.1175/JCLI-3284.1

[5] Carvalho, L. M. V., Jones, C., and Liebmann, B. (2002). Extreme Precipitation Events in Southeastern South America and Large-Scale Convective Patterns in the South Atlantic Convergence Zone. JOURNAL OF CLIMATE, 15, 18.

[6] Carvalho, L. M. V., Jones, C., and Liebmann, B. (2004). The South Atlantic Convergence Zone: Intensity, Form, Persistence, and Relationships with Intraseasonal to Interannual Activity and Extreme Rainfall. JOURNAL OF CLIMATE, 17, 21.

[7] Espinoza, J. C., Chavez, S., Ronchail, J., Junquas, C., Takahashi, K., and Lavado, W. (2015). Rainfall hotspots over the southern tropical Andes: Spatial distribution, rainfall intensity and relations with large-scale atmospheric circulation: Rainfall hotspots over the southern tropical Andes. Water Resources Research, 51(5), 3459–3475. https://doi.org/10.1002/2014WR016273

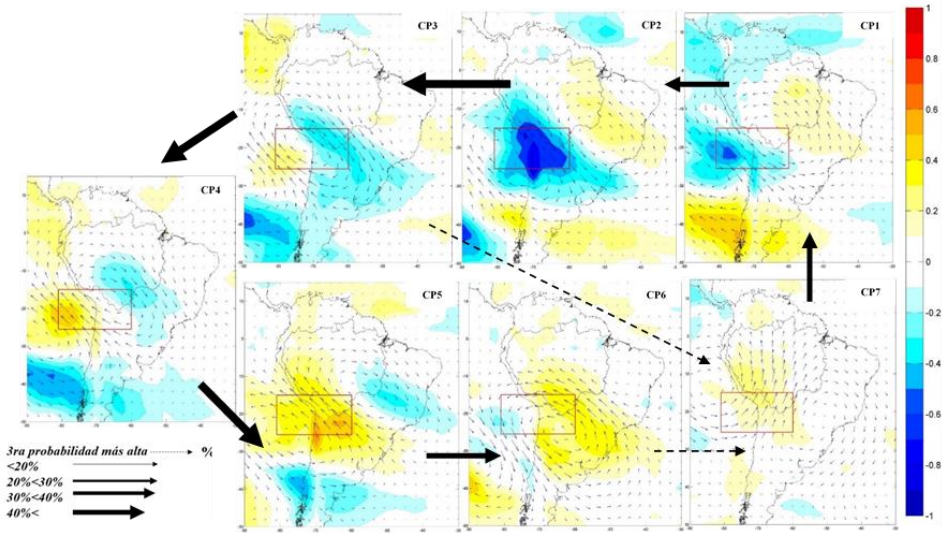


Figura 4: Composición de los patrones de circulación (CPs) denominada carta de circulación para el invierno (JJA) de anomalías estandarizadas de vientos a 200 hPa (vectores) y OLR (colores sólidos) para cada punto de grilla y con perido común de 1979 – 2013. Posibles sistemas convectivos presentes (ausentes) en cada patron están asociados a valores negativos (positivos) de OLR en color azul (rojo). Las fechas indican la probabilidad significativa de transición de un patrón a otro según la prueba t (ρ<0,1). El recuadro de color rojo representa el área donde han sido determinados los CPs.

[8] Espinoza, J. C., Garreaud, R., Poveda, G., Arias, P. A., Molina-Carpio, J., Masiokas, M., Viale, M., and Scaff, L. (2020). Hydroclimate of the Andes Part I: Main Climatic Features. Frontiers in Earth Science, 8, 64. https://doi.org/10.3389/feart.2020.00064.

[9] Espinoza, J. C., Lengaigne, M., Ronchail, J., and Janicot, S. (2012a). Large-scale circulation patterns and related rainfall in the Amazon Basin: A neuronal networks approach. Climate Dynamics, 38(1–2), 121–140. https://doi.org/10.1007/s00382-011-1010-8

[10] Espinoza, J. C., Marengo, J. A., Ronchail, J., Carpio, J. M., Flores, L. N., and Guyot, J. L. (2014). The extreme 2014 flood in south-western Amazon basin: The role of tropical-subtropical South Atlantic SST gradient. Environmental Research Letters, 9(12), 124007. https://doi.org/10.1088/1748-9326/9/12/124007

[11] Espinoza Villar, J. C., Ronchail, J., Guyot, J. L., Cochonneau, G., Naziano, F., Lavado, W., De Oliveira, E., Pombosa, R., and Vauchel, P. (2009). Spatio-temporal rainfall variability in the Amazon basin countries (Brazil, Peru, Bolivia, Colombia, and Ecuador). International Journal of Climatology, 29(11), 1574–1594. https://doi.org/10.1002/joc.1791

[12] Falvey, M., and Garreaud, R. (2007). Wintertime Precipitation Episodes in Central Chile: Associated Meteorological Conditions and Orographic Influences. Journal of Hydrometeorology, 8(2), 171–193. https://doi.org/10.1175/JHM562.1

[13] Ferreira, L. V., Venticinque, E., and Almeida, S. (2005). O desmatamento na Amazônia e a importância das áreas protegidas. Estudos Avançados, 19(53), 157–166.

[14] Francou, B., Rabatel, A., Soruco, A., Sicart, J.E., Silvestre, E.E., and Vuille, M.,. (2014). Glaciares de los Andes tropicales: Víctimas del cambio climático (Vol. 1). Banco Mundial.

[15] Gandu, A. W., and Silva Dias, P. L. (1998). Impact of tropical heat sources on the South American tropospheric upper circulation and subsidence. Journal of Geophysical Research: Atmospheres, 103(D6), 6001–6015. https://doi.org/10.1029/97JD03114

[16] Garreaud, René D. (2000). Intraseasonal Variability of Moisture and Rainfall over the South American Altiplano. MONTHLY WEATHER REVIEW, 128, 10.

[17] Garreaud, Rene D, and Aceituno, P. (2001). Interannual Rainfall Variability over the South American Altiplano. JOURNAL OF CLIMATE, 14, 11.

[18] Godoy, A. A., Campetella, C. M., and Possia, N. E. (2011). Un caso de baja segregada en niveles altos en el sur de Sudamérica: Descripción del ciclo de vida y su relación con la precipitación. Revista Brasileira de Meteorologia, 26(3), 491–502. https://doi.org/10.1590/S0102-77862011000300014

[19] Kohonen, T. (1988). An introduction to neural computing. Neural Networks, 1(1), 3–16. https://doi.org/10.1016/0893-6080(88)90020-2

[20] Kohonen, T. (2001). Self-Organizing Maps (3rd ed.). Springer-Verlag. https://doi.org/10.1007/978-3-642-56927-2

[21] Krishnamurti, T. N., Tewari, M., Chakraborty, D. R., Marengo, J., Dias, P. L. S., and Satyamurty, P. (1999). Downstream Amplification: A Possible Precursor to Major Freeze Events over Southeastern Brazil. WEATHER AND FORECASTING, 14, 30.

[22] Marengo, J. A., Liebmann, B., Grimm, A. M., Misra, V., Silva Dias, P. L., Cavalcanti, I. F. A., Carvalho, L. M. V., Berbery, E. H., Ambrizzi, T., Vera, C. S., Saulo, A. C., Nogues-Paegle, J., Zipser, E., Seth, A., and Alves, L. M. (2012). Recent developments on the South American monsoon system: RECENT DEVELOPMENTS ON THE SOUTH AMERICAN MONSOON SYSTEM. International Journal of Climatology, 32(1), 1–21. https://doi.org/10.1002/joc.2254

[23] Marengo, J., Cornejo, A., Satyamurty, P., and Nobre, C. (1997a). Cold Surges in Tropical and Extratropical South America: The Strong Event in June 1994. MONTHLY WEATHER REVIEW, 125, 28.

[24] Marengo, J., Cornejo, A., Satyamurty, P., and Nobre, C. (1997b). Cold Surges in Tropical and Extratropical South America: The Strong Event in June 1994. MONTHLY WEATHER REVIEW, 125, 28.

[25] Marengo, José A, Nicolini, M., and Douglas, M. W. (n.d.). identificación de casos de jets en niveles bajos, del norte (sallj) y del sur (sj) al este de los andes durante el verano y otoño 1999. 12.

[26] Marengo, Jose A, Soares, W. R., Saulo, C., and Nicolini, M. (2004). Climatology of the Low-Level Jet East of the Andes as Derived from the NCEP–NCAR Reanalyses: Characteristics and Temporal Variability. JOURNAL OF CLIMATE, 17, 20.

[27] Mayta, V. C., Silva, N. P., Ambrizzi, T., Dias, P. L. S., and Espinoza, J. C. (2020). Assessing the skill of all-season diverse Madden‐Julian oscillation indices for the intraseasonal Amazon precipitation. Climate Dynamics, 54(7‐8), 3729‐3749. https://doi.org/10.1007/s00382-020

[28] Paccini, L., Espinoza, J. C., Ronchail, J., and Segura, H. (2018). Intraseasonal rainfall variability in the Amazon basin related to large-scale circulation patterns: A focus on western Amazon-Andes transition region: intra-seasonal rainfall variability in western amazon. International Journal of Climatology, 38(5), 2386 ‐ 2399. https://doi.org/10.1002/joc.5341

[29] Sicart JE; Espinoza J; Queno L; Medina M. 2015. Radiative properties of clouds over a tropical Bolivian glacier: seasonal variations and relationship with regional atmospheric circulation. International Journal of Climatology, 36:3116-3128. Consultado 22 abr. 2017.

Machine learning to characterize thefts in Barranquilla City

Jhonny Castillo
Facultad de Ingeniería
Universidad Simón Bolívar
Barranquilla, Colombia
jonny.castillo@unisimonbolivar.edu.co

Paul Sanmartin Mendoza
Facultad de Ingeniería
Universidad Simón Bolívar
Barranquilla, Colombia
psanmartin@unisimonbolivar.edu.co

Diana Heredia
Facultad de Ingeniería
Universidad Simón Bolívar
Barranquilla, Colombia
dianahv@unisimonbolivar.edu.co

Daladier Jabba
Departamento de Ingeniería de Sistemas
Universidad del Norte
Barranquilla, Colombia
djabba@uninorte.edu.co

Elías Said Hung
Universidad Internacional de la Rioja
Madrid, España
elias.said@unir.net

***Abstract*— Data mining is an area of study in computer science, where statistical and mathematical techniques are applied to discover knowledge among a large amount of data. Data mining and other applications of artificial intelligence can contribute to understanding social and economic phenomena and contribute to propose creative solutions. This research aims to use Clustering data mining techniques and decision trees to geo-referenced data of thefts in a particular city in Colombia, occurring between the years of 2016 and 2017, to obtain a descriptive and predictive model. Information was obtained from open data (for free use for research purposes), getting a classification efficiency of 51% and 59.7%, respectively. Results can be considered not very good as predictors, but still interesting as descriptors and in manipulating geographic data, which are non-numeric attributes with many values.**

Keywords—Data mining; Geo-referenced data; Clustering; Decision trees

I. INTRODUCTION

Data mining is a set of data analysis techniques used for several years to extract information and valuable knowledge from a large amount of data about various sources, such as information systems, web, and mobile applications, physical documents, and surveys. The current proliferation of mobile communication devices with geographic positioning systems (GPS), available to people, has allowed the information to be enriched with this type of data.

The insecurity experienced in Latin America and the world has awakened in government entities and private sectors a motivation to follow up on events of crimes that have been historically recorded. According to this, in the Caribbean region of Colombia, a meticulous record is kept of many of the daily committed thefts. It appears that in 2016 in Barranquilla city, a register showed 2.297 reports of thefts in different modalities.

Regarding the cooperation of citizens and the authority's intervention, it is possible to have quick access to most data about crimes or thefts in social networks and applications for crowdsourcing, including data from the information systems of official entities such as the National Police, Attorneys General Office, and Institute of Legal Medicine. This information provided by the community is essential and considered the object of study for any type of research due to facilitating geo-referenced data besides the reporting of occurred events.

This research is one reason for increasing the motivation for exploring and applying big data techniques in geo-referenced data. Other motivations are described below:

The available data only contains geo-referenced information corresponding to the approximated point where the events occur. There is no complete information on the geographical area, and for that reason, particular techniques of spatial data mining cannot be applied. Another motivation is to seek a characterization of the theft data for Barranquilla to identify the characteristics and understand the phenomenon of theft in the city. The most important motivation of this work is to find a model for predicting the occurrence of crimes in the city of Barranquilla (Colombia) from the available data so that it is possible to replicate the methodology and techniques to other cities in the country.

A. The contributions of this work are the following:

- Basic data mining techniques have been applied to geographic data corresponding to single points (latitude and longitude).
- Geographically referenced data have been manipulated as real numbers to simplify their processing and subsequent interpretation of results.
- The occurrence of criminal acts (thefts) has been characterized by historical data.
- The occurrence of thefts in specific sectors of the city has been predicted, with an acceptable percentage of certainty (60%)

B. *Structure of this work*

The rest of the paper is divided into four sections: in section 2, related works are explained. Section 3 defines material and methods, section 4 describes the results, and finally, section 5 offers some concluding remarks.

II. Previous works

Extracting interesting and useful patterns from this type of data set is more complicated than removing numerical and categorical ones [1]. Two main approaches have been used: 1) The application of simple data mining algorithms after a pre-processing of spatial data (Using distance or topological approach, for example), and 2) the development of new algorithms that deal with spatial data; currently, there are several specialized data mining tools such as GeoMiner and GeoKmine, which are complicated to implement [2].

Previous works in spatial data mining are related to information in geographical databases, which store data as complex geometric objects like lines or polygons [2], [3], [4]. The primary data mining tasks applicable to this type of data are detection of dis-continuities and outliers, co-occurrence and co-location, spatial clusters and hotspots, and location prediction. Within the latter, the techniques commonly used are linear regression, Bayesian classifier, neural networks, and decision trees [3]. In general, previous works in spatial data mining use clustering (like k-means on pre-processed data), classification (like Naïve-Bayes), and association rules algorithms [2], [5], [6], [7]. Statistical tools are useful, such as likelihood hypothesis, probability, and spatial statistics [8].

Additionally, Hadoop-based services such as SpatialHadoop [9] and MapReduce framework [10] are used. Some authors have developed research that allows defining concepts such as Wu et al. in [11] and Binzani et al. [12]. Wu et al. research discusses the new development for spatial data mining technology. Barzani et al. introduce specific techniques such as Spatial association mining used to discover relationship patterns among spatial features based on spatial proximity from a sizeable spatial database.

Data mining on geo-referenced data has many applications such as location-based services, global change using satellite imagery, crime hotspots, traffic control, migration of birds, fishing control, and others [13], [14].

The questions then arise: ¿How to take advantage of geographic positioning data to predict the occurrence of certain types of events in specific places? and what kind of decisions can be made? From these questions, it was decided to research the application of data mining techniques to geo-referenced data (Geographical Coordinates), to obtain a predictive model of criminal facts. The occurrence of thefts in Barranquilla (Colombia) was taken as a case study in this research.

The following sections describe the work carried out, the results obtained, and the conclusions are presented.

III. Materials and Methods

The first step was to obtain a set of freely accessible data containing geo-referenced data. The geo-referenced data is the geographical position of a point in terms of longitude and latitude. For this study, data about thefts in Barranquilla (A city in Colombia) and its metropolitan area was given by the Fiscalía General de la Nación (Colombia). The original data set had about 28,000 examples with 24 attributes, corresponding to 2016 and 2017; the file format was MS Excel. Many of the records contained blank, incomplete, or missing data. This incomplete information was eliminated to reduce the probability of biases or errors during the execution of the data mining processes.

Some of the columns or fields contained non-contributory values, such as "Country," which for all records is "Colombia." Other fields had no unique values, such as the address of the place of events or many missing values. For that reason, it was decided to ignore these attributes. Subsequently, empty rows (or almost empty) were verified and deleted, assuming that they were the result of incomplete or erroneous information in the original system. The mentioned before was developed to avoid biases in the data. As an additional process, points with coordinates in the dataset were visualized in a map identifying outliers. These points with coordinates out of the study were also deleted. At the end of the cleaning process, there were 21,720 records and 10 attributes in the data set; they provide information about different dimensions of the situations: Spatial (Latitude and longitude), Temporal (Day, month, year, day-of-the-week, date, and time) and Circumstantial (Mode and Characterization).

TABLE I. Definitive attributes of the data set.

Attribute	Description	Values	Discretization
Latitude	Geographic coordinate	Min: 13.313 Max: 11.055 (numeric)	2 bins
Longitude	Geographic coordinate	Min: -75.234 Max: -74.74 (numeric)	2 bins
Day	Day value in the timestamp	1 – 31 (numeric)	No
Month	Month value in the timestamp	January – December (nominal)	No
Year	Year un the timestamp	2016 – 2017 (numeric)	No
Day-of-the-week	Day of the week corresponding to the timestamp	Sunday – Saturday (nominal)	No
Time	Hour: minute: seconds in timestamp	00:00:00 – 23:59:59 (Time)	Time slots, 4 hours each of them
Modality	Mode of the theft	3 different values (nominal)	No
Characterization (Target / classifier)	Type of theft	6 different values (nominal)	No

The class attribute (target) was the one called "Characterization," which contains the type of crime reported: 1. theft to people, 2. residences, 3. financial institutions, 4. vehicles, and 5. others (not specified in original data). The fig. 1 shows the distribution of those values. The three highest values are theft to people, to vehicles, and to financial institutions.

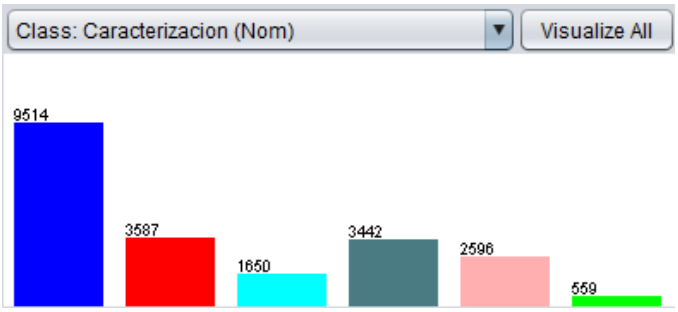


Fig. 1. Distribution of the values of Characterization attribute.

Through the visualization of coordinates in QGIS (Fig. 2) and Google Earth (Fig. 3), erroneous points and points outside the study's geographical area of interest were detected. The conglomeration of points on the map corresponds to Barranquilla and its metropolitan area.

The following task was to discretize the numeric attributes with continuous values, including the latitude and longitude attributes taken as real numbers of values. It was unsupervised executed in 2, 3, and 4 bins, and then 2 was chosen as the best. Attributes like day-of-the-week and month were not discretized; the attribute time was discretized in time slots, 4 hours each one. Table 1 shows the resulting attributes used.

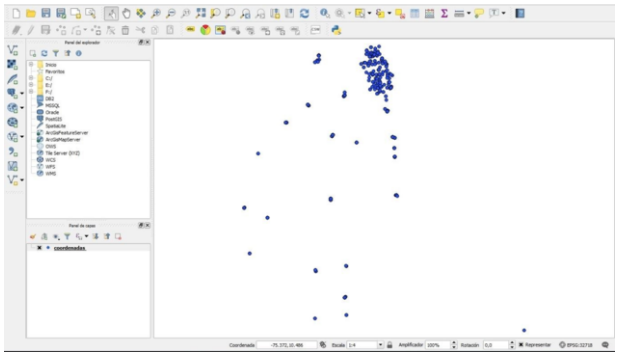


Fig. 2. QGIS points

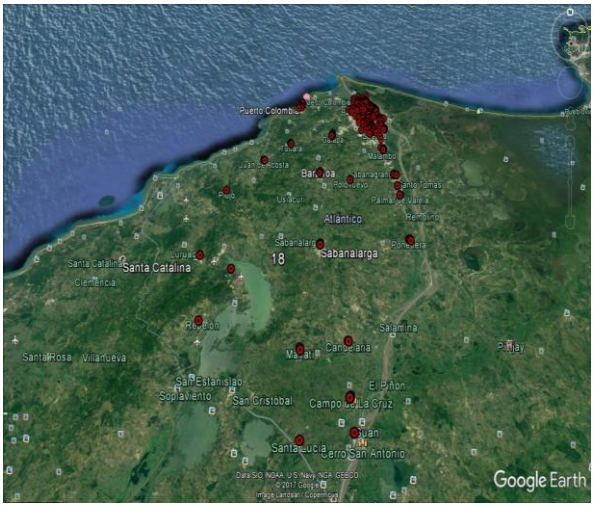


Fig. 3. Visualization of coordinates using Google Earth.

The data mining software tool used is WEKA, an open-source machine learning soft-ware containing multiple built-in tools and standard machine learning tasks, and it does not need programming. The Machine Learning Group developed it at the University of Waikato in New Zealand.

Once the dataset was ready for mining, it was chosen the techniques to be applied: clustering and decision trees. In WEKA, algorithms were K-means clustering and the J48 decision tree. Clustering is the division of the dataset into object groups based on the similarity among their values. It can be viewed as a data modeling technique providing concise summaries of the data [15]. K-means was selected because is the most used clustering method and easy to understand the results. It divides the dataset into k disjunctive clusters; its objective function is to assess the partitioning quality. Objects within a cluster are like one another, dissimilar to objects in other clusters [16]. A decision tree is a flowchart-like tree structure, where each internal node (non-leaf node) denotes a test on an attribute, in which each branch represents an outcome of the test, and each leaf node (or terminal node) holds a class label [16]. The J48 tree in Weka is an implementation of the C4.5 tree, in which each node evaluates a single attribute and derive the confidence interval from the training data. The resultant tree is easy to understand [17].

During the execution of the mining process, multiple tests were carried out with the chosen techniques. The first purpose was to determine the influence of the variables on the obtained models and take the decision to exclude some of them. Subsequently, to establish the best discretization in the variables that required it.

The algorithms were executed using n-fold cross-validation and n = 5, which split the data set into n subsets of equal size, utilizing each subset for testing and the remainder for training. The learning algorithm was applied to n different training sets. The estimated errors are averaged to produce an overall estimated error, and the standard deviation is calculated. The value of n = 5 was chosen because greater values did not significantly improve [17].

The metric to evaluate the classifier performance is the confusion matrix, which contains four blocks:

- True positives (TP): Positive tuples correctly labeled by the classifier.
- True negatives (TN): Negative tuples correctly labeled by the classifier.
- False positives (FP): The negative tuples incorrectly labeled as positive by the classifier.
- False negatives (FN): The positive tuples incorrectly labeled as negatives.

The confusion matrix is a useful tool for analyzing how well the obtained classifier is. The higher values are in TP and TN, better is the classifier [16].

IV. Results

All the executions of the data mining process were made with the same data set. After one implementation, it was

evaluated the efficacy of the technique in terms of "Incorrect classified (or clustered) instances." At the end of each test, it was decided to make changes in the discretization or ignore any of the columns. The results were recorded and documented to establish a comparison subsequently. The conditions of the executions carried out, and the results obtained are explained below, in the next subsections.

A. *First test*

In this first attempt, six attributes were used because these contain important and useful information about phenomenon in this study: Characterization (Target), Modality, Month, Day-of-the-week, Latitude, and Longitude. K-means clustering was run with 2 clusters and evaluated with the class attribute (Target), obtaining 60% incorrectly clustered instances. One of those clusters had more than 80% of the total data, and their centroids had three attributes with equal values, as shown in Fig 4. This figure represents modality, latitude, and longitude.

With the same dataset, the J48 tree was executed using 5 folds of cross-validation and 2/3 of the data to train the model. The obtained model showed the modality in its root. It means that it is the most contributing attribute, but it has many different values. For that reason, the tree was highly sparse. Its efficacy was 57% (instances correctly classified), a better result than clustering; however, the confusion matrix showed that many of the theft types were misclassified, like "theft to people," as shown in Fig. 5. In addition, the RMSE (Root Mean Squared Error) was 0.3091 and the MAE (Mean Absolute Error) was 0.1909.

```
KMeans
=====


Number of iteration: 3
Within cluster sum of squared errors: 42926.0

Initial starting points (random) :

Cluster 0: ATRACO,'\'(-74.987087-inf)\'' ,'\'(10.683823-inf)\ '' , 'marzo   ' , ' sábado   '
Cluster 1: ATRACO,'\'(-74.987087-inf)\'' ,'\'(10.683823-inf)\ '' , 'marzo   ' , ' martes   '

Missing values globally replaced with mean/mode

Final cluster centroids
                                                    Cluster#
Attribute                        Full Data                 0                 1
                                 (21348.0)          (17145.0)          (4203.0)
==============================================================================
Modality                         ASSAULT            ASSAULT            ASSAULT
Longitude                 '(-74.987087-inf)'  '(-74.987087-inf)'  '(-74.987087-inf)'
Latitude                   '(10.683823-inf)'   '(10.683823-inf)'   '(10.683823-inf)'
Month                            december           december           august
DayWeek                          saturday           saturday           tuesday
```

Fig. 4. Centroids of clusters were obtained in the second test.

```
==== Confusion Matrix ===

   a    b    c    d    e    f  <-- classified as
9185    0   85    4  183   57 |  a = THEFT TO PEOPLE
2104 1479    2    0    0    2 |  b = VEHICLE THEFT
 763    2  851    3    0   31 |  c = THEFT TO RESIDENCES
2696    0  357  323    0   66 |  d = THEFT TO FINANCIAL ESTABLISHMENTS
2322    0    0    1  254   19 |  e = THEFT OF CELL PHONES
 362    0   36    8    4  149 |  f = OTHERS
```

Fig. 5. Confusion Matrix for J48 in the first test.

B. *Second test*

The values of the attribute named modality were grouped in 4 categories, and the Characterization attribute was grouped in 2: "Theft to Person" and "Others," since almost 50% of the data are in the first one. This group was made previously, in excel, and clustering and J48 were executed with the same six attributes used before. The obtained clusters had centroids with three attributes with the same values, like the previous documented test. Despite the above described, the model received 47% of the instances misclassified.

Using J48 obtained a very small tree that takes only modality, latitude, and longitude. It acquired 59% of the instances correctly classified. However, the MAE was 0.2667 and the RMSE was 0.3665. Fig. 5 shows the resulting tree.

C. *Third test*

Based on former results, it was decided the following:

- It was removed the instances with latitude and longitude outlier values, which were detected after a specific analysis in these two attributes. It was discarded 2000 rows approximately.
- Latitude and longitude were not discretized.
- Characterization and modality continued with grouping values as the last test.

The algorithms were executed under these conditions, using 5 folds of cross-validation, and splitting the data set in 66% for training and the remaining for testing. K-means obtained 48% incorrectly clustered instances with better cluster characteristics, as presented in Fig 6. The J48 achieved 61% correctly classified and a tree with 181 leaves.

```
Final cluster centroids:
                                          Cluster#
Attribute                 Full Data              0             1
                          (19276.0)      (8439.0)     (10837.0)
==============================================================
Modality                  ASSAULT        ASSAULT        ASSAULT
Longitude                   -74.8047     -74.8016      -74.8072
Latitude                     10.9681      10.8016       10.9823
Month                     december       april        december
DayWeek                   saturday       saturday     friday
```

Fig. 6. Final cluster centroids obtained in the third test.

D. *Fourth test*

It was decided to group the values of the classifier attribute into two categories, using MergeInfrequentNominalValues filter in Weka and executing clustering with 3, 4 and 5 clusters with the aim of achieving separate clusters that improve the classification of instances. Fig. 7 shows the final cluster centroid executed for 3 clusters. Most of the centroids in all the cases have modality “Assault”, the value in amost 60% of the records in dataset.

Final cluster centroids:

Attribute	Full Data (21348.0)	Cluster# 0 (8721.0)	1 (4514.0)	2 (8113.0)
Modality	ASSAULT	ASSAULT	ASSAULT	ASSAULT
Longitude	-74.8106	-74.8127	-74.8286	-74.7948
Latitude	10.9442	10.9879	10.8741	10.9361
Month	december	december	december	August
DayWeek	saturday	saturday	tuesday	saturday

Fig. 7. Final cluster centroids obtained with 3 clusters in the fourth test.

The k-Means algorithm was executed using 66% of the instances for training. The classification testing in each case obtained the following results:

- With 3 clusters, 59.64% of the instances were incorrectly clustered.
- With 4 clusters, 71.4% of the instances were incorrectly clustered.
- With 5 clusters, 74.9% of the instances were incorrectly clustered.

In all cases, the centroids of the clusters indicate some separation between them, but the classification was not useful. It can be inferred that any theft can happen in any city area, and there seems to be no way to obtain clusters with a greater degree of differentiation between them.

E. Fifth test

This final testing was executed with the following conditions:

- The original values of characterization and modality were pre-processed using MergeInfrequentNominalValues filter in Weka, which grouped the values with less than 5000 occurrences in the data set, obtaining 2 classes for each attribute.
- It was removed instances due to outlier values in latitude and longitude attributes.

The algorithms were executed with the same parameters as the last test. It was obtained better results: 61.16% correctly classified in J48 and 51.3%, with k-means. Fig 8 shows the confusion matrices. Despite the improvement in the global indicator, the confusion of the J48 classifier persists, being greater in theft from persons. The clusters resulting from this test present even worse classification confusion than the J48 tree (See Fig. 9).

```
==== Confusion Matrix ===

   a    b   <-- classified as
1778 1230 |   a = THEFT TO PEOPLE
1297 2249 |   b = OTHERS
```

Fig. 8. Confusion Matrix for J48, fourth test.

```
==== Confusion Matrix ===

   0    1   <-- assigned to cluster
3740 5187 |   a = THEFT TO PEOPLE
4699 5650 |   b = OTHERS

Cluster 0 <--  OTHERS
Cluster 1 <--  THEFT TO PEOPLE
```

Fig. 9. Confusion Matrix for k-means, fourth test (Classification executed with complete dataset).

V. Conclusion

In the study case, the application of clustering and decision trees on geo-referenced data obtained results moderately satisfactory. The best result of J48 as a predictor was 61.16%. Despite the numerical result, using these techniques produced a relevant description of the situation, such as the type of crime with more incidence in some places in the city or on certain days of the week. For example, in the case of clusters, the geographical position and the modality were the same for all of them in all the tests carried out, but with different times (month and day of the week); it did not obtain disjoint clusters, which lead to misclassification.

The values of the attribute named modality were grouped in 4 categories, and the Characterization attribute was grouped in 2: "Theft to Person" and "Others," since almost 50% of the data are in the first one. This group was made previously, in excel, and clustering and J48 were executed with the same six attributes used before. The obtained clusters had centroids with three attributes with the same values, like the previous documented test. Despite the above described, the model received 47% of the instances misclassified.

The handling of geographical position data (longitude and latitude) as numerical values simplifies its interpretation. Still, it is also reduced to the management of vast areas, not specific sites, due to the discretization necessary for this type of data. This would lead to thinking about the possibility of establishing geographic areas of adequate size to group specific sites. Likewise, it is necessary to keep the value of longitude and latitude of a point as a single data, not two individual ones, as they were managed in this study.

Non-numerical attributes initially have many different values; the techniques used are not efficient in handling this data, making its reduction necessary, using only the highest occurrence values, and grouping the remaining ones into a single value. This form of a decrease in values may implicitly lead to a bias, as it is perceptual.

Advancing these types of proposals will help the development of new monitoring techniques around the progress of crimes. At least potentially, it can benefit in better development of mechanisms for measuring and monitoring the policies and security measures put in place by local, regional, and national public authorities; as well as the follow-up of it, by social actors interested in the oversight and impact of crime in some geographical regions. The study can also improve citizen security by identifying data mining techniques such as

those proposed in this work, which may contribute to the active monitoring of criminal actions carried out at the social level.

REFERENCES

[1] L. Hamdard; A. Abdaoui; N. Belattar and M. Alchikha, EasySDM: An integrated and easy to use Spatial Data Mining platform, de 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K), Lisbon, 2015.

[2] A. Kumar; A. Kakkar; R. Majumdar and A. Singh, Spatial Data Mining: Recent Trends and Techniques, de International Conference on Computer and Computational Sciences (ICCCS), 2015.

[3] J. Refonaa; M. Lakshmi and V. Vivek, Analysis and prediction of natural disaster using spatial data mining technique, de International Conference on Circuit, Power and Computing Technologies [ICCPCT], 2015.

[4] D. Cangrejo and J. Agudelo, Minería de datos espaciales, Avances en Sistemas e Informática, vol. 8, nº 3, pp. 71-77, 2011.

[5] B. He; Y. Cui; J. Chen and P. Xie, A spatial data mining method for mineral resources potential assessment, de Proceedings 2011 IEEE International Conference on Spatial Data Mining and Geographical Knowledge Services, 2011.

[6] A. Garaeva; F. Makhmutova; I. Anikin and K. Sattler, A framework for co-location patterns mining in big spatial data, de 2017 XX IEEE International Conference on Soft Computing and Measurements (SCM), 2017.

[7] D. Singh, S. Tanweer y T. Nafis, «Scalable Geo Web Techniques using Spatial Data Mining for Air Pollution,» International Journal of Advanced Research in Computer Science, vol. 8, nº 3, pp. 1160-1166, 2017.

[8] E. Ahmed et al., "The role of big data analytics in Internet of Things," Computer Networks, vol. 129, pp. 459-471, 2017.

[9] Vaishali Sontakke; R B Dayanand, Optimization of Hadoop MapReduce Model in Cloud Computing Environment. 2019: IEEE, pp.

[10] Q. Wu, Y. Wang, F. Liu, and Y. Xia, "Knowledge Discovery in Power System Based on Spatial Data Mining," 2017: IEEE, pp. 330-334.

[11] K. Binzani and J. S. Yoo, "Spark-based Spatial Association Mining," 2018: IEEE, pp. 5300-5301.

[12] V. Borgony and S. Shekhar, Spatial and Spatio-temporal Data Mining, de IEEE International Conference on Data Mining, 2011.

[13] P. Sanmartin, V. Quintero, D. Heredia, R. Romero and J. Castillo "Datos, información, tendencias, tres miradas sobre un contexto cambiante". Ediciones Universidad Simón Bolívar, ISBN: 978958-5533-08-0. 2018.

[14] P. Berkhin, A Survey of Clustering Data Mining Techniques, de Grouping Multidimensional Data, Berlin, Springer, 2006, pp. 25-71.

[15] J. P. J. and. K. M. Han, Data mining: Concepts and Techniques, Elsevier, 2011.

[16] I. Witten; E. Frank; M. Hall and C. Pal, Data Mining: Practical Machine Learning Tools and Techniques, Morgan Kaufmann, 2017.

Fuzzy Logic Based Agent for Flood-Awareness

Jose Simmonds
Departamento de Informática
Universidad Carlos III de Madrid
Leganes, Spain
jose.simmonds@alumnos.uc3m.es

Juan A. Gómez
Departamento de Biología Marina
Universidad de Panamá
Panama, Panama
juanay05@hotmail.com

Agapito Ledezma
Departamento de Informática
Universidad Carlos III de Madrid
Leganes, Spain
ledezma@inf.uc3m.es

***Abstract*—Today's climatic conditions worldwide are escalating, particularly in hurricanes and extreme rainfall, being the latter, the trigger for severe flood inundation problems. Therefore, this is a situation of high concern specifically for catchments in the tropical regions and this represents the reasons an expert fuzzy system is vital in flood-awareness in the tropical scenario to determine the decision-making responsibility for proper control of the input comprising rainfall, river stage, and streamflow. However, as the implementation and maintenance of such systems comes with their cost and hydrometric data, when not available, poses serious problems of forecasting and the creation of robust inference systems. So said, this approach is an initiative to provide users and the layperson to get information of floods which if substantially disclosed by combining data mining to fix data issues, artificial intelligence and soft computing for knowledge extraction, and agent technology for autonomous handling of the system in one prototype model can save lives and expect the destruction of property.**

***Index Terms*—Agent, fuzzy logic, flood-awareness, inference system, multi-agent archetype**

I. Introduction

Given their limitations and failures to operate properly [1], the need for continuous implementation of flood-awareness systems, the creation and optimization of existing strategies and plans to mitigate the effects caused by flood, continues to increase in importance and increase the state of awareness among water science and hydrological engineering professionals, governmental officials, stakeholders, watershed managers and the general population on a global scale, and in particular in tropical regions in the face of the accelerated increase in climatic phenomena.

The climate continues to change, and the hydrologic cycle is shifting for sure. Severe flooding is a current increasing problem that represents one of the major causes of catastrophic preeminence to civilian and livestock detriment, the economy, agriculture, and damage to property in Panama. The Panamanian reality shows little or the absence of studies for both flood forecasting and the application of early warning systems for dimensions of time in advance. Most of the relevance in the development of such systems arises from data availability problems because of the poor to non-existing hydrometric stations in the catchment.

Due to the growing paradigms of data mining (DM), artificial intelligence (AI), and soft computing (SC), to mention a few, the gaps in the studies of hydrological phenomena, caused by the scarce information related to hydrometric data, have been gradually filled. Thus, recently, the techniques presented by fuzzy logic have been successfully applied to modeling and obtaining inferences from hydrological databases with certain vague, distorted, and deficient data. Therefore, it is a challenging pursuit to design a flood-awareness system for tropical catchments that suffers from severe hydrometric data deficiency, unless these deficiencies are accurately addressed through data reconstruction techniques [2].

In the present manuscript, we exhibit the use of the fuzzy-based system supported on the Mamdani fuzzy inference system and applied it to a reconstructed hydrometric dataset to disclose the information it contains for use in flood-awareness knowledge.

II. Literature Review

A. Fuzzy Logic in Flood-Awareness Systems

For the last decades, the development of early warning systems has had a growing interest in certain research fields among the scientific and engineering communities to address advanced situations of natural disasters such as floods caused by intense rainfall, hurricanes, and climatic events others such as earthquakes, volcanic eruptions, and tsunamis, which represent a threat to both the life of civilians and wildlife alike, as well as to impairment to water quality in watercourses, damage to the economy, and infrastructure. Early awareness systems also find very important use in problems of urban transportation flow, vehicle collisions, the outbreak of a disease. As mentioned earlier, flood is one of the worst disaster challenges in Panama.

Since its first development by the pioneering works of Lofti Zadeh [3], [4], fuzzy logic continues to find its rightful place in helping to identify and respond to questions that are common to systems presenting imprecise and vague information. To this extent, let us skim through a few works in which fuzzy logic has shown success to the main areas of concerns in hydrology, like flow modeling [5]– [8], flood control [9], [10], and presently with some research done for flood warning system based on agent technologies such as agent-based modeling and simulation (ABMS) and multi-agent systems (MAS) archetypes, as documented in [11], [12]. We also can find a thorough up-to-date paper on fuzzy logic applications in hydrology presented in the following review [13]

The coupling of fuzzy logic with other AI methodologies, as shown to exhibit greater strength in handling hydrologic data.

However, a few approaches and applications are employing the former techniques towards the role of solving hydrologic problems related to floods, as discussed in [14]. In this work, we harness the potential of fuzzy inference system (FIS) simulation, which is based on fuzzy logic, to infer knowledge from a hydrometric data set whose deficiencies we corrected beforehand with data mining approaches.

III. Experimental Settings

A. Multi-agent System Approach

Depicted in Fig. 1 is a schematic representation of the recommended rationale for the fuzzy logic prototype that we are presenting here as a component (i.e., the classification level) of our proposed MAS for flood forecasting in the tropical river basin [14]. Therefore, this fuzzy logic skill is an element that is embedded in an agent forming part of this multi-agent system dynamics, and consequently as the role of decision agent. Therefore, from the system frame of reference, suggested in the flood forecasting system composed of ten main agents, is as follows.

- Rainfall Sensor Agent (AgentRNSn) The key task of the AgentRNSn is to receive and aggregate the rainfall readings retrieved from the precipitation gauge.
- Water Level Sensor Agent (AgentWLSn) The key task of the AgentWLSn is to receive and aggregate the river stage readings retrieved from the stage gauge or level radar.
- Streamflow sensor agent (AgentSFSn) The key task of the AgentSFSn is to receive and aggregate the river flow readings retrieved from the flow meter.
- Sensor verification agent (AgentSV) The SV is the sentinel for the health of the sensors in the hydrometric sensors network. It requests from the AgentHDBM the storage of the hydrometric sensor raw readings into the database. It reports with the AgentUI on sensors functionality and also communicates with the hydrometric sensor agents mentioned previously.
- Sensor Data Preprocessing Agent (AgentDPP) The main function of the AgentDPP is to optimize the raw data, by performing data imputation of missing values.
- Hydrodatabase Management Agent (AgentHDBM) The main function of the AgentHDBM is to store the raw data the preprocessed data, the flow forecasting results, alerts, the user's subscriptions, and to dispatch information on data to other agents.
- Agents Classifier and Fuzzy Logic (AgentCLA and AgentFL) Two forecaster agents, an AgentForecaster1 and an AgentForecaster2 whose main roles are to perform flood forecasting estimates on a time horizon of one, two, three, or four hours lead time and the AgentFL as the role to perform flood-awareness inference concerning the hydrometric inputs and, on the information, delivered by the two forecaster agents.
- User interface agent (AgentUI) The responsibility of the AgentUI is to coordinate the user-mas-system interactions.

B. Recommended Rationale and Fuzzy Logic Prototype

Haven presented a synthesized explanation of the proposed MAS entities formulation, it should be noted, that in this paper, we are only interested in presenting the overall interaction that entails the role of the fuzzy logic agent concerning in due time, the usage of the real-time data stream, and that of the forecasting results from the Forecaster1 and Forecaster2 agents for simulating the flood-awareness. However, we must make it clear that now, and for testing purposes, the AgentFL is only employing the hydrometric data information directly from standalone files and not data fed into the system in real-time. For example, the rainfall agent, stage (water level) agent, and river flow agent, are connected to the field sensors in real-time, each of which senses the surroundings of a hydrometric station for perceptions on precipitation, river water level, and flow.

The capturing of the crisp input variables collected by the hydrometric sensors agents becomes conceptualized because of the fuzzification mechanisms into linguistic quantities which are in direct relation to the input linguistic variable quantity. The fusion of the rainfall, stage and streamflow data will become earmarked for the decision agent to infer the flow regime and deliver warning and disclosure of information to the users concerning the measure to be adopted.

C. The Fuzzy Model Layout

The design for this fuzzy logic task is implemented with the three input hydrometric variables which we have initially divided into four linguistic variables depending on the type of variable (see Table I). In this sense, for these hydrometric variables, the linguistic partitions are as follows: i) rainfall, with linguistic variables Light, Average, Intense, VeryIntense, ii) variable stage, Normal, Average, High, VeryHigh and iii) the variable streamflow, Low, Average, High, and VeryHigh. For each of these, its degree of membership is also identified within the fuzzy set. There are several membership functions (MFs) that can build fuzzy inference system such as the Triangular, Trapezoidal, Gaussian, Sigmoidal [15]. We used the triangular membership function (Fig. 2) given its ease of use in handling accurate ranges and its evaluation is much simpler than the other contra parts, that even the lay on non-technical user can easily understand it. The calibration and tuning process of fuzzy rule systems, along with the identification and selection of suitable MFs, although they are several insights and approaches offered in the literature to conduct such task, it can be a tedious one and totally depend on the problem to be solved. However, fuzzy rule-based systems can be enhanced with adequate calibration techniques to improve the efficiency of the model output. Perhaps with the tuning process, the fuzzyfication process, with the conjunction of an expert's input, is helpful in the formation of fuzzy rules. The tuning process depends on the set rule base define for the system and the database. The tuning process we followed here is the well-known *ad hoc* data covering technique proposed in [16]. An iterative method

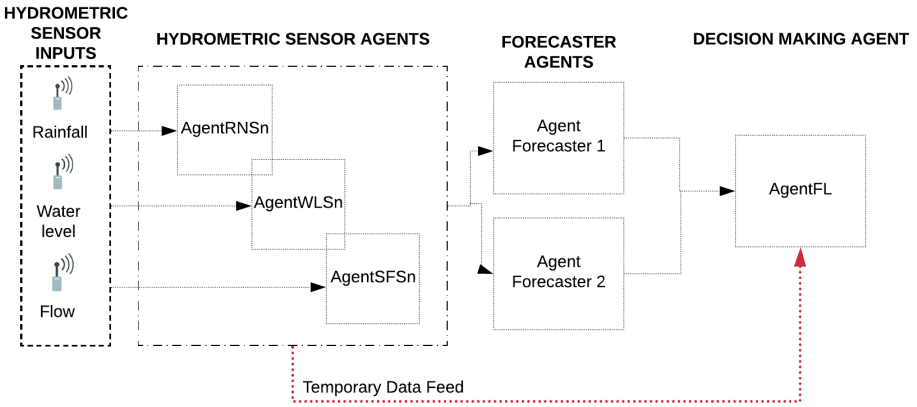


Fig. 1. A representation of the recommended fuzzy logic model rationale, showing the real-time data input level, the hydrometric data level, classifier level, and the corresponding decision-making level.

which as a low time-consuming process for learning of the fuzzy rules by covering the criteria of the data modeled.

TABLE I
INPUT AND OUTPUT VARIABLES FOR THE FLOOD AWARENESS DATA

Hydro Variable	Linguistic Tag	Range
Rain (mm)	Light	[1 – 50]
	Average	[25 – 75]
	Intense	[50 – 150]
	VeryIntense	[100 – 250]
Stage (m)	Normal	[8.0 – 9.0]
	Average	[8.5 – 10.0]
	High	[9.5 – 11.0]
	VeryHigh	[10.5 – 12.0]
Streamflow (cms)	Low	[0.5 – 100]
	Average	[50 – 200]
	High	[100 – 300]
	VeryHigh	[200 – 400]
Flood-Awareness	LOW	[0 – 4]
	CAUTION	[2 – 6]
	ALARM	[4 – 8]
	RISK	[6 – 10]

IV. Remarks and Discussion

Once the configurations of the fuzzy system have been set up with the input variables of interest, and the rules have been established, the previous are duly merged with their convenient linguistic variables and hence the inference engine, using the Fuzzy Logic Toolbox of the MATLAB computational package (2020b, Academic version) accesses the fuzzy rule base to compute and generate both the average and the output linguistic values (Fig. 3). Hence, before this process can take place, there are two important steps to complete which lead to these processes. These steps are the aggregation and composition phases. With aggregation, the output for each fuzzy set rule is integrated with the inference phase. In simpler

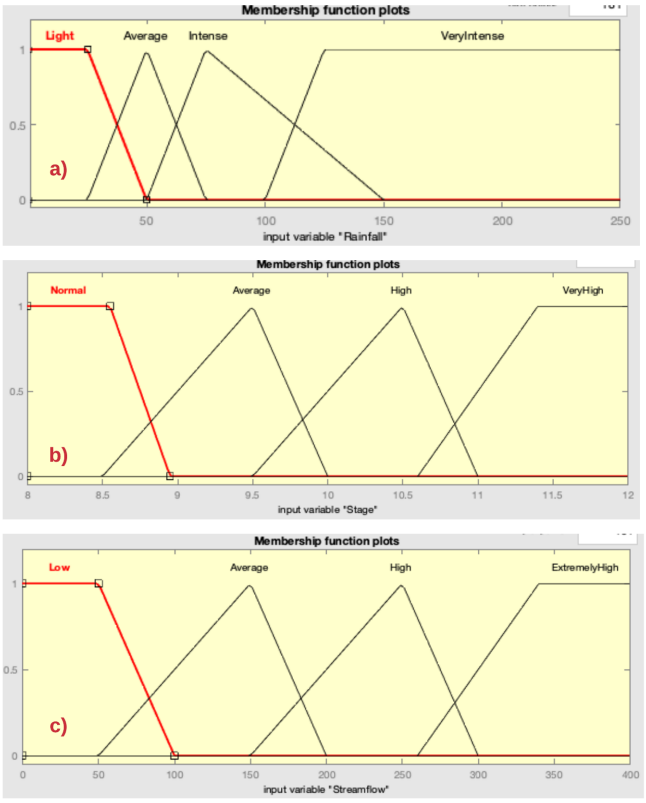


Fig. 2. Triangular membership function for three hydrometric input values: a) rainfall, b) stage, c) streamflow.

terms aggregation computes the IF portion of the rule whereas in the composition phase, the THEN operation is computed.

The fuzzy inference structure is the central unit of a fuzzy rule implementation from linguistic variables upon which ex-

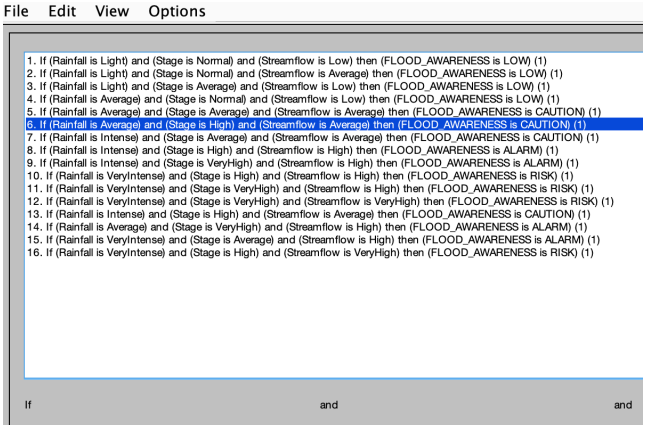


Fig. 3. Fuzzy rules implementation.

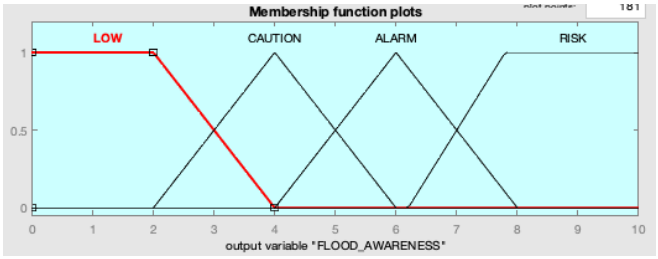


Fig. 4. Flood-awareness universe of values.

perts base their knowledge on decision making. It relies on the ”IF-THEN” rules alongside the ”OR” or ”AND” connectors for representing necessary decision rules. In this experiment, a set of sixteen rules were created as can be seen in Fig. 4. The decision rules were generated from the three hydrologic input parameters (rainfall, stage level and streamflow) and according to these is decided the output flood-awareness level value. The system was simulated using the min-max or maximum means (MOM) Mamdani's model and the centroid (COG) method for defuzzification. Therefore, if the actual number of active rules are p^n, being p the maximum number of membership functions and n the number of inputs, the design for the fuzzy system based on the three mentioned inputs above is 3 inputs with 4 membership each, which results in a total of $3 \times 4 = 12$ rules. However, to enhance the system further, and safeguard a workable setup of fuzzy rules and membership functions be valid and practical in computational terms, especially for real-time operation purposes as the suggested by this system, four more rules where added based on expert opinion and knowledge of the precipitation regime and conditions of local flow for the river system within the catchment of study.

During the defuzzification mechanism, the values corresponding to the outputs of the linguistic variables are transformed into their crisp format. Thus, in the arrangement for each linguistic term, its maximum value, in turn, represents the maximum (center) values of each membership function. This means that the maximum value of the interval of a membership function is given by the median of the interval. Now, regarding

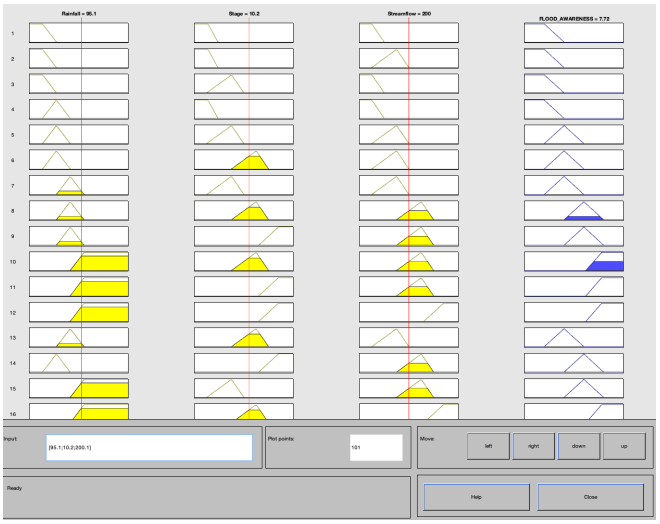


Fig. 5. Fuzzy inference simulation output.

the final decision concerning the flood-awareness state that the system should disclose, the usual values for the linguistic terms are given by ”LOW”, ”CAUTION”, ”ALERT”, and ”RISK”, as described in Figs. 3 and and 4 respectively. In this way, the best combinations of the input values are calculated using the weighted average and considering their degrees of membership. Once these calculations are carried out, the decision agent (i.e., fuzzy logic agent) based on the results of the linguistic terms oversees announcing to the user interface agent about the flood-awareness levels defined in Table I. As the three hydrometric agents are constantly perceiving the hydrologic conditions of the environment, to evaluate the performance of the fuzzy logic decision-maker agent, the situation of the simulation of the changes in flooding severity is shown by the mapping from streamflow, stage, and rainfall to the flood-awareness level (Fig. 4).

The above system has been applied to infer the flood-awareness levels of the Medio River in Panama. The total drainage area of the Medio sub-catchment is 15 km^2, the total length of the stream is about 50,000 m, and the maximum elevation difference is 80 m. The average annual rainfall is 3,000 mm and it is allocated evenly throughout the year. The annual evaporation is approximately 1,100 mm.

The hydrometric data set, from 2012-2016 consists of hourly and fifteen minutes precipitation, water level, and streamflow readings measured during low and high flow periods from an installed rain gauge, flow meter, and surface water elevation radar transmitted in real-time. From this dataset, we decided to use the period corresponding to November 2012, as our case study to build and test the system, since it is a period that presented the highest rainfall, stage height, and discharge. Then, to test the system, for example, if we evaluate on five rule cases, as shown in Fig. 3 according to the MATLAB simulation, the outputs consider the accumulated flood volume in the channel, giving the input precipitation and the water

TABLE II
OUTPUTS ON FLOOD AWARENESS INFERENCE BASED ON THE HYDROMETRIC DATA INPUTS

Cases	Hydro Variable			Flood Awareness Level
	Rainfall (mm)	Stage (m)	Streamflow (cms)	
1	15.1	8.9	55.8	1.8
2	9.1	8.5	57.8	1.6
3	40.5	9.2	105.4	4.0
4	95.1	10.2	200.0	7.7
5	169.1	12.0	389.6	8.5

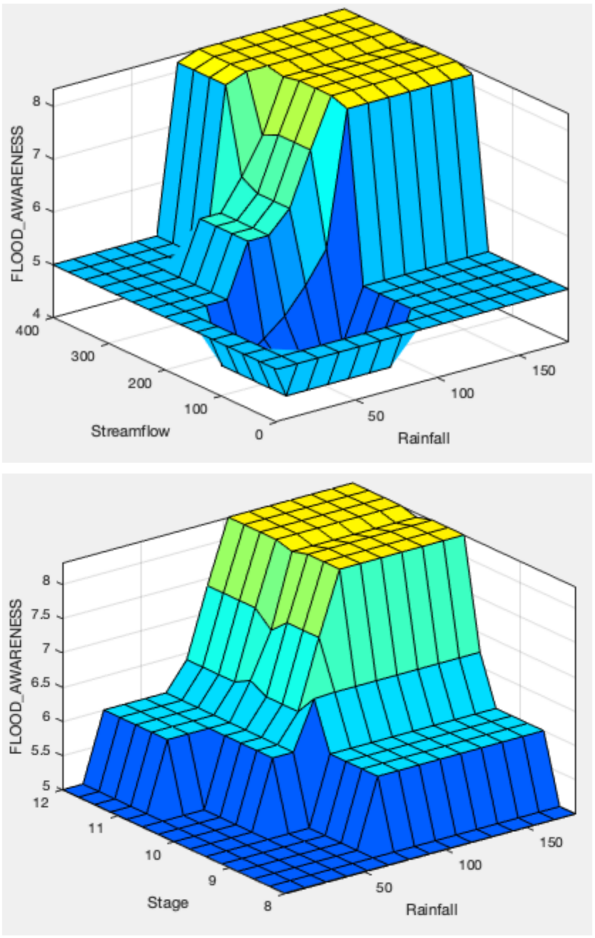


Fig. 6. A spatio-temporal representation of the results of the simulated membership functions related to the three inputs and the fuzzy inference output.

level to hydraulically reproduce the final decision of the fuzzy agent model, as it reproduces the flood-awareness as expected (see Table II). Therefore, we can see that when the system is simulated for a given range of input values, say for example, if the value of rainfall is = 96.1 mm, which lies within the range 50-150 as intense, and stage = 10.2 m, which lies within the range 9.5-11.0 as high, and the streamflow = 200 $m^3 \cdot s^{-1}$, which lies within the range 100-300 as high, the resulted flood-awareness level is 7.7 (Fig. 4). Therefore, high values of flood-awareness, for high values of rainfall, streamflow, and water level, and low values of the flood-awareness for low values of rainfall, water level, and streamflow as compared with Table II and Figs. 5 and 6 respectively.

V. CONCLUSIONS AND FUTURE WORK

In the present work, we have presented the fuzzy logic component of a multi-agent system model proposed for flood forecasting in the tropical river basin. This model, besides having a component for decision-making, based on a fuzzy agent with capabilities to infer flow conditions in the river, also comprises other modules with capabilities for managing and storing hydrometric data. The experiment that was discussed in this manuscript showed that the hydrometric variables of rainfall, water level, and streamflow, once implemented with their membership function and crisp inputs and subjected to the process of fuzzification and defuzzification methods and validated by certain rules that express the participatory capacity building of an expert domain, represents a highly effective tool when dealing with data that present high uncertainty. Hence, for each condition of the hydrometric values, the system could infer the conditions of the flow regime in the channel as was shown in Table II.

For continuing the work, it is required we adjust the fuzzy sets further, incorporate the flow predictions results given by the two forecaster agents for inputs to the AgentFL, and that we expect the complete MAS model implemented with fuzzy logic and other artificial intelligence tools completed, updated, and extended to provide field deployment and by such provide the functionalities for flood-awareness inference.

ACKNOWLEDGMENT

This work has been partially supported by the Spanish Government under projects PID2019-104793RB-C31 and RTI2018-096036-B-C22. Likewise, the authors would like to thank the staff of the Vicerectoría de Investigación (Universidad de Panamá), Universidad de Panamá, Departamento de Biología Marina y Limnología, and the Secretaría Nacional de Ciencia, Tecnología e Innovación (SENACYT) for supporting this research project.

REFERENCES

[1] S. Ayeb-Karlsson, D. Kniveton, T. Cannon, K. van der Geest, I. Ahmed, E.M. Derrington, E. Florano, and D.O. Opondo, I will not go, I cannot go: cultural and social limitations of disaster preparedness in Asia, Africa, and Oceania, Disasters, 43(4), pp.752–770.

[2] J. Simmonds, J. A. Gómez, and A. Ledezma, Data Preprocessing to Enhance Flow Forecasting in a Tropical River Basin. InInternational Conference on Engineering Applications of Neural Networks 2017 Aug 25 (pp. 429–440). Springer, Cham.

[3] L.A. Zadeh, Fuzzy sets. Information and control, vol. 8 (1965), pp. 338–353.

[4] L.A. Zadeh, Fuzzy sets as a basis for a theory of possibility. Fuzzy sets and systems, 100, 1999, pp. 9–34.

[5] E.D.P. Perera, and L. Lahat, Fuzzy logic based flood forecasting model for the Kelantan River basin, Malaysia. Journal of hydro-environment research. 2015 Dec 1;9(4):542–53.

[6] N.K. Kulkarni, and V.V. Shete, "Hybrid neuro-fuzzy approach for flood prediction and Dam gate control." In International Conference on Information Society (i-Society 2014), pp. 213–218. IEEE, 2014.

[7] S. Supatmi, R. Hou, and I.D. Sumitra, Study of Hybrid Neurofuzzy Inference System for Forecasting Flood Event Vulnerability in Indonesia. Computational intelligence and neuroscience, 2019.

[8] D. Han, I.D. Cluckie, D. Karbassioun, and J.A. Lowry, fuzzy logic approach to river flow modelling. InStochastic Hydraulics 2000 2020 Dec 17 (pp. 853–860). CRC Press.

[9] K.S. Desai, and M.V.D. Bakarola, Intelligent Flood Control System Using Fuzzy Logic.

[10] Y. Waghmare, P. Choudhary, M. Rochlani, and M. Rajput, Fuzzy Logic Based Flood Warning and Floodgate Control System.

[11] T.N.M. Aris, H. Hamdan, N.C. Pa, and S.A. Nazeer, An agent-based ontology fuzzy logic conceptual model for flood warning prediction. In: Academics World 12th International Conference, 20 Dec. 2015, Singapore. (pp. 36–39).

[12] T.N.M. Aris, H. Hamdan, N.C. Pa, and S.A. Nazeer, Fuzzy Logic Model for Flood Warning Expert System Integrating Multi-Agent and Ontology. Journal of Engineering and Applied Sciences, 12: (2017) 3124–3129.

[13] S. Kambalimath, P.C. Deka, A basic review of fuzzy logic applications in hydrology and water resources. Applied Water Science. 2020 Aug;10(8):1–4.

[14] J. Simmonds, J. A. Gómez, and A. Ledezma, The role of agent-based tmodeling and multi-agent systems in flood-based hydrological problems: a brief review. Journal of Water and Climate Change, 11(4), 2020, pp. 1580–1602.

[15] S.N. Sivanandam, S. Sumathi, and S.N. Deepa, Introduction to fuzzy logic using MATLAB (2007). Berlin Heidelberg: Springer.

[16] L. X Wang, and J. M. Mendel, Generating fuzzy rules by learning from examples. IEEE Transactions on Systems, Man, and Cybernetics 22(6), 1992, pp. 1414-1427.

Fuzzy Logic Approach for Energy Poverty Identification in the Urban Regeneration Areas of Getafe

Ascensión López Vargas
Control Learning and Optimization Group, *Computer Sciences Department*
University Carlos III of Madrid
Leganés, Spain
aslopez@inf.uc3m.es

Raúl Castaño Rosa
Appropriate Technology for Sustainable Development Group
Electrical Engineering Department
University Carlos III of Madrid
Leganés, Spain
rcastano@ing.uc3m.es

Jorge/Martínez-Crespo
Appropriate Technology for Sustainable Development Group
Electrical Engineering Department
University Carlos III of Madrid
Leganés, Spain
jorgemar@ing.uc3m.es

Ulpiano Ruiz-Rivas Hernando
Appropriate Technology for Sustainable Development Group
Thermal and Fluids Engineering Department
University Carlos III of Madrid
Leganés, Spain
ulpiano@ing.uc3m.es

Agapito Ledezma Espino
Control Learning and Optimization Group,
Computer Sciences Department
University Carlos III of Madrid
Leganés, Spain
ledezma@inf.uc3m.es

***Abstract*—In this work, a conceptual framework based on Fuzzy Logic for identifying Energy Poverty situations in Getafe (Spain) is presented. The resulted model uses twelve input variables for obtaining the level of Energy Poverty. The input variables were defined around the three key drivers commonly used for the Energy Poverty characterization: low incomes, high energy costs, and poor thermal efficiency of buildings. Six Energy Poverty application domains were presented. Seven output levels were defined for a well-fitted characterization of the Energy Poverty situation in Getafe**

Keywords—Energy Poverty, Fuel Poverty, Artificial Intelligence, Fuzzy, Innovation, Sustainable Cities.

I. INTRODUCTION

The New Urban Agenda, adopted in Quito, Ecuador, on 20 October 2016, laid out standards and principles for the planning, construction, development, management, and improvement of urban areas working as an accelerator of the United Nations Sustainable Development Goal (SDG) 11 [1]: Make cities and human settlements inclusive, safe, resilient, and sustainable. In alignment with these programs, the Urban Innovative Actions (UIA) of the European Commission was created for providing urban areas throughout Europe with resources to test innovative and unproven solutions for tackling urban challenges [2]. In this respect, the UIA "EPIU Getafe Hogares Saludables" [3] aims to identify and reduce Energy Poverty (EP) in Getafe, a large southern city of Madrid (Spain), by applying new technologies such as Internet of Things (IoT) and Artificial Intelligence (AI). In Getafe, up to 30% of people are affected by EP and there are 156 recipients of the Integration Minimum Income for every 10,000 inhabitants [3]. Numbers get even worse in the "Las Margaritas" and "La Alhóndiga" neighborhoods of Getafe (both declared as Urban Regeneration Areas by the Government of Spain [4]): nearly the 50% of homes do not have heating systems and around 235 families demand for assistance to pay for basic supplies and rent [3]. The ultimate goal of this work is to develop a Fuzzy Logic based conceptual framework for identifying EP situations in the urban regeneration areas of Getafe based on criteria selected.

II. GETAFE ENERGY POVERTY MODEL

This section comprises the first approach to give a preliminary picture for characterizing the EP problem in Getafe. According to the Policy Report of the European Commission "Energy poverty and vulnerable consumers in the energy sector across the EU: analysis of policies and measures" [5], EP is characterized by three key drivers: low incomes, high energy costs, and poor thermal efficiency of building [5]. However, other aspects such as political, social circumstances or harsh weather conditions also play a high influential role in a situation of EP [6]. As a direct consequence of the complexity of the phenomenon, researchers have not been able to define the universal factors that affect EP [7], studying, in most cases, partially the problem by focusing on some of its aspects. In this work, the EP problem of Getafe was evaluated from 6 different aspects using 12 variables selected by EP experts as result of the study of the Getafe EP situation carried out though data (provided by the Getafe Municipality) and surveys.

Getafe is in southern Madrid, Spain, at 622 m.a.s.l. with a maximum average temperature of 32ºC in summer and minimum average temperatures at nearly 1.5 ºC in winter. The yearly average is 14.9ºC [8]. Six key aspects, according to [9], were selected for the EP characterization of Getafe: a) thermal comfort, b) electrical consumption, c) economic situation, d) domestic hot water system, e) electrical infrastructure and f) hidden EP situation. The low thermal comfort, the ambient temperature (inside the home), the energy consumption for heating, and the building energy efficiency were used to evaluate the thermal comfort of the dwellings. The building energy efficiency was determined based on the windows quality (degree of insulation provided and thermal/light characteristics), the insulation of the building and the presence of dampness, leaks, and rot. The electrical consumption aspect includes two variables: the energy bill (without heating) and the energy consumption per month. The monetary situation of the families was also evaluated through the monthly income and the delays in the energy bills payment. The performance coefficient was used for obtaining the adequate level of the Domestic Hot Water

(DHW) system access [10]. Regarding the electrical infrastructure, the lighting efficiency of the dwellings (presence/absence of LED lighting and natural light) as well as the domestic appliance efficiency class [10] were used. The latter was obtained by combining the efficiency of four typical appliances: TV, fridge, dishwasher, and washing machine [11]. The identification of hidden EP situations was obtained from personal evidence on how people perceive their own situation. Table I shows the summary of the conceptual model designed for Getafe including all variables selected as well as their range values.

TABLE I. CONCEPTUAL ENERGY POVERTY MODEL OF GETAFE.

EP Application domain	*Variables*	*Range values*
Thermal comfort	Thermal comfort	0-10
	Ambient temperature	18-30ºC
	Energy consumption for heating	200-1400 kWh
	Building efficiency	Evaluation of the windows quality, thermal characteristics and building conservation.
Electrical consumption (no heating included)	Consumption	100- 400 kWh
	Expenditures	Average 60€
Economic situation	Income	500-4500€
	Delays in the energy bills payment	Evaluation of the presence of delay
DHW system	Adequate supply system	Coefficient of Performance 0-5
Electrical infrastructure	Outdated or inefficient lighting	Evaluation of the absence of led lighting and natural light
	Outdated domestic appliances	Evaluation of the number of domestic appliances with high energy efficiency qualification
Hidden energy poverty	Hidden energy poverty auto evaluation	0-10

III. FUZZY LOGIC APPROACH FOR CHARACTERIZING THE ENERGY POVERTY IN GETAFE

In recent years, a huge number of Artificial Intelligence applications focused on partial aspects of EP or on areas intimately related to EP have been published [12,13]. However, there is a lack of works that apply AI exclusively to identify and/or alleviate EP. In this work, this research gap was filled and a conceptual framework for addressing the EP problem of Getafe by means of AI techniques (Fuzzy Logic based systems) was developed. Fuzzy logic has a form of human expertise by applying IF-THEN rules; there are no numerical values [14]. The use of this AI technique allows to use more flexible limits for the EP characterizing, especially with the input variables. Figure 1 includes the scheme of the Getafe EP fuzzy logic system. The architecture of the fuzzy system comprises three main modules: the fuzzifier, the interference engine and the defuzzifier. The first module (Fuzzification Module) aims to fuzzifier the input variables (Table I) by converting crisp input values to fuzzy values. After that, the Interference Engine evaluates the EP fuzzy values in the rule base designed and combines the results of each rule. Finally, the Defuzzification Module converts the output data to non-fuzzy values. The main core of the fuzzy logic are membership functions [14] used in the fuzzification/defuzzification modules for mapping the non-fuzzy input values to fuzzy linguistic terms and vice versa.

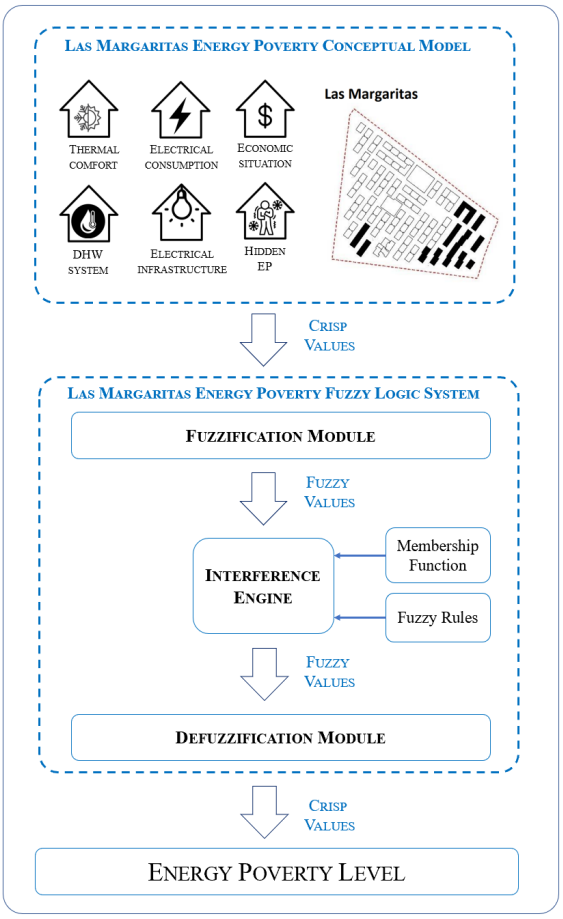


Figure 1: Scheme of the Getafe EP fuzzy logic system.

The Fuzzy modeling (including the Fuzzy Inference System, the Membership Functions, and the Fuzzy Rules) was implemented in the MATLAB environment, using the special module Fuzzy Logic Toolbox [15]. Figure 2 shows the final scheme of the system. The output (the level of EP: EP_Level) was defined as a combination of the twelve input variable levels after passing through the interference engine.

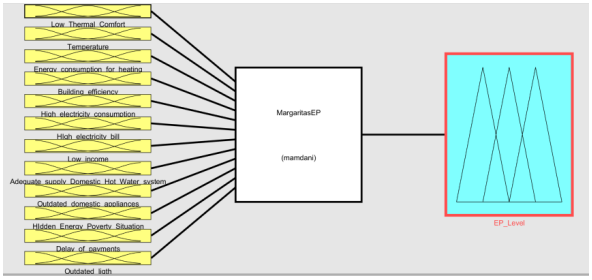


Figure 2: Fuzzy Logic scheme for characterizing the EP in Getafe using Implemented in MATLAB.

A. *Fuzzification Module*

In the fuzzy logic application, input variables were defined. Gaussian, "S" and "Z" functions were selected for the membership functions. Table II comprises the list of the twelve variables fuzzified as well as their values; the level of each variable was defined according to the fuzzy sets defined in Table I.

TABLE II: GETAFE ENERGY POVERTY FUZZIFICATION OF INPUT VARIABLES.

Variables	***Fuzzy set***
Thermal comfort	Dissatisfied, Somewhat satisfied, Satisfied, Very satisfied
Ambient temperature	Low, Good, High, Very high
Energy consumption for heating	Low, Good, High, Very high
Building efficiency	Very deficient, Deficient, Efficient, Very efficient
Consumption	Low, Good, High, Very high
Expenditures	Low, Normal, High
Income	Very low, Low, Good, Notable, Very good.
Delay in the payment of energy bills	Delay/no delay
Adequate supply system	High efficiency, Moderate efficiency, Low efficiency
Outdated or inefficient lighting	Very bad, Bad, Good, Very good
Outdated domestic appliances	Low efficiency, Medium efficiency, High Efficiency
Hidden energy poverty auto evaluation	You feel you need aid; you are satisfied with your situation; you are very satisfied with your situation

B. *Interference Engine*

The Interference Engine includes the cause-and-effect relationships relating the input and output variables. The rules were created with the IF – THEN. The Interference Engine employed was a Mamdani fuzzy inference system. Eq. 1 shows the output of the Interference Engine. The 7 outputs were selected by the EP experts; they cover all EP situations identified in Getafe.

$$Output \begin{cases} Severe\ EP \\ HIgh\ EP \\ Moderate\ EP \\ Low\ EP \\ Very\ Low\ EP \\ Vulnerability \\ No\ EP \end{cases} \quad (1)$$

For modeling the relationship between the membership functions, a list with more than fifty fuzzy rules was created providing an EP level as a result of the input variables combination. For designing, first antecedents were evaluated, and fuzzy operators were applied. Then, the implications were identified (the level of EP).

C. *Defuzzification Module*

The output of the fuzzy system was obtained as the output of the Defuzzification Module. Seven levels of EP were defined: Severe, High, Moderate, Low, Very Low, Vulnerability and No Energy Poverty. This module is designed by transforming the fuzzy output into crisp values (as the EP percentage between 0-100%). Centroid method was used for defuzzification. Figure 3 shows the membership functions of the output defined; gaussian functions were used.

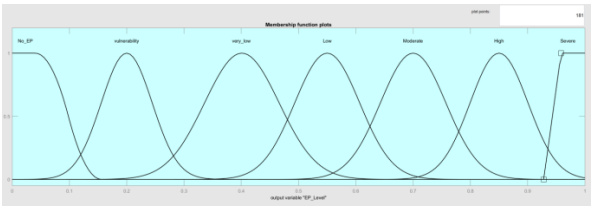


Figure 3: Membership functions of the output variable.

Figure 4 shows the 3D diagram of the designed system (a no EP case is evaluated).

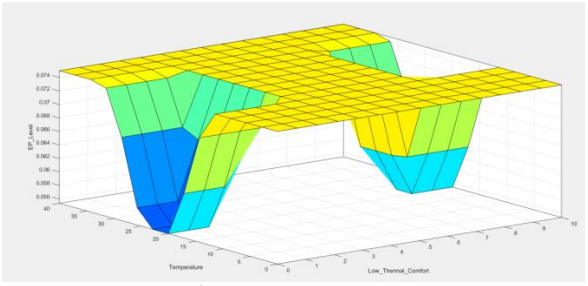


Figure 4: 3D model of the EP system (no EP case: 8 25 600 8 225 70 4300 5 9 8 0 9).

IV. RESULTS

For evaluating the model, various cases of EP were studied. Based on [16], ten families' situations with different EP profiles were evaluated: three families with no EP and seven cases with different levels of EP. Table III shows the values of the three representative EP cases employed for evaluating Getafe EP fuzzy system corresponding to No EP, High EP and Severe EP situations.

TABLE III. NO EP, HIGH EP AND SEVERE EP CASES USED FOR EVALUATING GETAFE EP FUZZY SYSTEM.

Variables	***No EP***	***High***	***Severe***
Thermal comfort	9	1	1
Ambient temperature	22	14	12
Energy consumption for heating	500	800	100
Building efficiency	9	3	1
Consumption	300	300	49
Expenditures	65	45	70
Income	1800	600	400
Delays in the energy bills payment	no	yes	yes
Adequate supply system	5	1	1
Outdated or inefficient lighting	9	1	1
Outdated domestic appliances	9	1	1
Hidden energy poverty auto evaluation	9	1	1

The output of the Getafe EP fuzzy system was the level of EP in percentage (EP_Level variable). In the case of the "No EP" situation, the output of the system was EP_Level=7.5%. The output of the system was EP_Level=64.13% for the "High" EP case. The "Severe" EP situation gave as an output EP_Level=83.74%. EP experts have evaluated the results and it was confirmed that the system could be used for identifying EP cases in Getafe.

V. CONCLUSIONS

A fuzzy model for EP identification was designed and developed for estimating the EP level in Getafe. Monetary, social, and energy information was used as input variables, while the level of EP was used as an output variable of the system. More than fifty rules were implemented for estimating the different EP levels. Although this first approach must be carefully considered, authors think that it can be seen as a comprehensive response to understanding different levels of EP at the neighborhood level. The system was a Mamdani type and Centroids was the method used in the defuzzification process; the system was implemented in MATLAB. As a future line of research, the system should identify the level of EP with an improved level of accuracy and, additionally, set the type of corrective solutions to be applied for each EP situation.

ACKNOWLEDGMENT

The authors acknowledge funding from the European Commission, "Urban Innovative Actions", EPIU Getafe Project, UIA04-212.

REFERENCES

[1] United Nations website. Accessed: May 27, 2021. Available online < https://sdgs.un.org/goals/goal11>

[2] Urban Innovative Actions website. Accessed: May 27, 2021. Available online < https://www.uia-initiative.eu/en >

[3] Hogares Saludables website. Accessed: May 27, 2021. Available online < https://hogaressaludables.getafe.es/en/>

[4] Getafe Municipality website. Accessed: May 27, 2021. Available online < https://www.getafe.es/getafe-amplia-a-649-las-viviendas-que-seran-rehabilitadas-con-las-arru-en-las-margaritas-y-la-alhondiga/ >

[5] "Energy poverty and vulnerable consumers in the energy sector across the EU: analysis of policies and measures" Report. European Comission website. Accessed: May 27, 2021. Available online < https://ec.europa.eu/energy/sites/ener/files/documents/INSIGHT_E_Energy%20Poverty%20-%20Main%20Report_FINAL.pdf>

[6] Castaño-Rosa, Raúl & Barrella, Roberto & Sánchez, Carmen & Barbosa, Ricardo & Kyprianou, Ioanna & Paschalidou, Eleftheria & Thomaidis, Nikolaos & Dokupilová, Dušana & Gouveia, João & Kadar, Jozsef & Hamed, Tareq & Palma, Pedro. (2021). Cooling Degree Models and Future Energy Demand in the Residential Sector. A Seven-Country Case Study. Sustainability. 13. 2987.

[7] Filčák, Richard, and Lidija Živčič. "Energy Poverty and Multi-Dimensional Perspectives of Social Inequalities and Policy Challenges." International Issues & Slovak Foreign Policy Affairs, vol. 26, no. 1-2, 2017, pp. 40–61. JSTOR.

[8] Climate Data website. Accessed: May 27, 2021. Available online < https://es.climate-data.org/europe/espana/comunidad-de-madrid/getafe-885499/>

[9] Castaño-Rosa R, Solís-Guzmán J, YRubio-Bellido C, Marrero M, "Towards a multiple-indicator approach to energy poverty in the European Union: A review," Energy and Buildings, Volume 193, 2019, Pages 36-48, ISSN 0378-7788.

[10] JMM-SAT-Ferroli website. Accessed: May 27, 2021. Available online <https://www.jmmsatferroli.com/etiqueta-certificacion-energetica/>

[11] IDAE website. Accessed: May 27, 2021. Available online https://www.idae.es/uploads/documentos/documentos_Documentacion_Basica_Residencial_Unido_c93da537.pdf

[12] Shahaboddin Shamshirband, Dalibor Petković, Rasul Enayatifar, Abdul Hanan Abdullah, Dušan Marković, Malrey Lee, Rodina Ahmad, Heat load prediction in district heating systems with adaptive neuro-fuzzy method, Renewable and Sustainable Energy Reviews, Volume 48, 2015, Pages 760-767, ISSN 1364-0321.

[13] Papada L, Katsoulakos N, Doulos I, et al. Analyzing energy poverty with fuzzy cognitive maps: a step-forward towards a more holistic approach. Energy Source B Econ Plan Policy 2019; 14: 1–24.10.1080/15567249.2019.1634162.

[14] Kanika Mittal, Amita Jain, Kunwar Singh Vaisla, Oscar Castillo, Janusz Kacprzyk, A comprehensive review on type 2 fuzzy logic applications: Past, present and future, Engineering Applications of Artificial Intelligence, Volume 95, 2020, 103916, ISSN 0952-1.

[15] Mathworks website. Accessed: May 27, 2021. Available online <https://es.mathworks.com/products/fuzzy-logic.html>

[16] Castaño-Rosa R, Solís-Guzmán J, Marrero M. A novel Index of Vulnerable Homes: Findings from application in Spain. Indoor and Built Environment. 2020;29(3):311-330.

Clustering energy-poor households using machine learning and socioeconomic indicators

Miguel Núñez-Peiró
School of Architecture
Universidad Politécnica de Madrid
Madrid, Spain
miguel.nunez@upm.es

José Antonio Iglesias Martínez
CAOS Group
Universidad Carlos III de Madrid
Madrid, Spain
jiglesia@inf.uc3m.es

Ana Sanz-Fernández
School of Architecture
Universidad Politécnica de Madrid
Madrid, Spain
ana.sanz@upm.es

Agapito Ismael Ledezma Espino
CAOS Group
Universidad Carlos III de Madrid
Madrid, Spain
ledezma@inf.uc3m.es

Carmen Sánchez-Guevara Sánchez
School of Architecture
Universidad Politécnica de Madrid
Madrid, Spain
carmen.sanchezguevara@upm.es

Marta Gayoso Heredia
School of Architecture
Universidad Politécnica de Madrid
Madrid, Spain
marta.gayoso@upm.es

***Abstract*— Energy poverty is a widespread multidimensional phenomenon that affects a significant share of European households. Although a variety of methodologies are available for the detection of energy poverty, they often merely distinguish between households that are energy poor and those that are not, failing to analyse the different sub-groups that might exist and whose needs and possibilities for intervention may be highly variable. In this study, households in energy poverty from the Community of Madrid (Spain) have been sub-clustered and characterised. We use the Household Budget Survey to determine which households lie under energy poverty and divide them into three main groups. Then, we use the k-means method to subcluster each energy poverty group from a set of 50 socio-economic indicators. Finally, decision trees are employed to study the characteristics of each subcluster. Results yield a total of 12 subclusters distributed within the three energy poverty main groups. Economic activity-related indicators seem to be the most relevant for explaining the sub-clustering (up to 95% correctly classified instances), although other indicator's categories proved to be relevant as well (e.g., housing characteristics). Results also show the potential of this process to identify the main drivers of energy poverty as well as the key elements that configure each subcluster of vulnerable households, which are expected to help fine-tune intervention policies.**

Keywords—energy poverty, fuel poverty, decision analysis, households budget survey, machine learning, decision support tool

I. INTRODUCTION

In order to define energy poverty, several approaches have been developed during the last three decades. One of the first definitions was Boardman's, in which *a household suffers from fuel poverty when it is unable to have adequate energy services for 10% of household income* [1]. Closely related to this definition is the income and expenditure approach, widely used to identify whether or not a household is considered to be in energy poverty. Other methods, such as the consensual approach, are also common to address the detection of energy poverty. Despite all of them being convenient to point out the existence of energy poor households, a more in-depth description of the energy poor could be done.

Energy poverty could be caused by low-income levels, high energy needs, high energy cost or a combination of them, and identifying the cause or causes is key to address the problem. In addition, it also can take place in a wide range of different contexts and household situations which differences may lead to diverse solutions. For example, as pointed out in some studies [2], pensioners, unemployed, those with low levels of education, women as main breadwinners or tenants are more vulnerable to suffer energy poverty. Nevertheless, a step forward to identify how these households are grouped is needed. The grouping of these characteristics may help not only to better describe the problem and identify the causes, but also to recommend specific and adapted solutions to each cluster.

Digitalization and the technologies associated with this process are causing a rapid economic and social transformation at a global level. Artificial Intelligence (AI), perhaps the most disruptive technology at the moment, plays a fundamental role in this transformation. In recent years, AI has experienced a rise not only in terms of research but also in its applicability. The growth that developments in AI are experiencing is due, on the one hand, to the increase in the processing capacity of computers, the improvement of algorithms and the exponential growth in the volume and variety of digital data [3]. According to Gartner [4], Artificial Intelligence is *a technology that appears to emulate human performance, typically by learning, coming to its own conclusions, appearing to understand complex content, engaging in natural dialogues with people, enhancing human cognitive performance, or replacing people in performing non-routine tasks.*

In the last decade, AI has boosted research work in practically all areas of knowledge using general-purpose techniques and methods. Nowadays, the use of AI techniques is becoming essential in all areas of knowledge since they improve the performance of other traditional techniques and processes, which implies, among other things, improving and facilitating research tasks. AI encompasses numerous techniques and methods ranging from the well-known Expert Systems, Machine Learning and Evolutionary Algorithms to the, lately so current, Deep Neural Networks.

This paper proposes the use of Machine Learning techniques for clustering energy-poor households. It applies AI to subcluster previously identified energy poverty household groups, analyses the outcomes and characterises the energy-poor households subclusters using different indicators. To do that, and using the specific domain knowledge, the following objectives were covered:

XXX-X-XXXX-XXXX-X/XX/$XX.00 ©20XX IEEE

- To select the appropriate indicators for clustering.
- To group the indicators for building decision trees.
- To analyse each subcluster identifying the proper indicators for that end.
- To characterize each subcluster highlighting their features and differences.

II. Methodology

A. *Identifying energy poor households*

The methodology presented in this paper uses the Spanish Household Budget Survey (HBS)[5] (the data set for 2019 is the one used for the analysis). Although this database was not expressly designed for energy poverty detection, it is broadly used for that purpose partially due to the lack of a more specific survey on this topic. The HBS, updated on an annual basis, includes information regarding household incomes and expenditure on energy bills, which can be used to identify the effort made by the households to meet their energy needs. In addition, the database includes other characteristics of the households such as family composition, activity and/or employment of the main breadwinner and the rest of the family members, characteristics of the main breadwinner (such as gender, education level and age) and housing stock characteristics (year of construction, size, type, heating systems availability, tenure status, etc.). The microdata of this database, with a specific data set for each respondent household, is available. Thus, data-crossing for all the characteristics of every household is possible. The total sample size in the Community of Madrid (the case of study in this research) is 345 households. Each of this household has a demographic elevation factor, which may vary for each of them. These factors reflect how many households are represented by each household and with the use of different elevation factors the entire population of the region is represented.

The methodology explored for the clustering is based on the energy-poor household's classification developed by Sánchez-Guevara et al. [6] and recently incorporated in the last update of the long-term Strategy for Energy Rehabilitation in the Building Sector in Spain [7]). This methodology classifies households according to their income and energy expenditure. On that base, households are classified following these thresholds:

- The energy poverty line is defined as twice the median value of the last five years' median energy expenditure in Spain [8]. This value is around the 10% of the income.
- The energy poverty vulnerability line is fixed as half the energy poor threshold (5%).
- The monetary poverty baseline is set, in compliance with Eurostat Methodology, as the 60% of the median income.
- Monetary vulnerability stands between monetary poverty and the median income value.

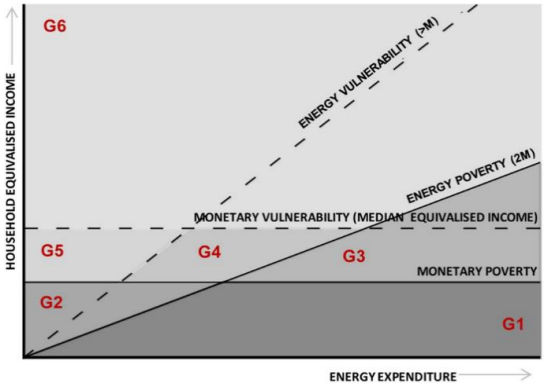


Fig. 1. *Schematic Energy poverty evaluation chart* [2]

Based on these thresholds, households are classified as follows:

Group 1 (G1): these households are located below the monetary poverty line and below the energy poverty line, due to their high energy expenditure.

Group 2 (G2): these households' income is below the monetary poverty line but above the energy poverty line. Previous research has demonstrated these households are at risk of energy poverty as they are likely suffering from inadequate temperatures due to the lack of heating services (Sánchez-Guevara et al., 2015).

Group 3 (G3): these households are considered energy-poor, being below energy poverty line but over the monetary poverty line.

Group 4 (G4): these households are considered vulnerable, given their close position in regard to both energy and monetary poverty thresholds.

Group 5 (G5): these households are vulnerable towards monetary poverty due to their close position to this threshold.

Group 6 (G6): these households are considered not vulnerable.

In this research, the groups G1, G2 and G3, which correspond to those households that are more likely to suffer from energy poverty, were selected for a detailed analysis.

B. *Subclustering energy poor households*

Once the formerly explained division among households into different groups is conducted, the next step would be to identify whether those groups are homogeneous or there are different subclusters within them. The indicators related to income and expenditure were the ones used to classify the groups in the first place and using those for sub-clustering could be considered redundant. On the other hand, the other features present in the database (such as family composition, activity and/or employment of the main breadwinner and the rest of the family members, characteristics of the main breadwinner and housing stock characteristics) are key to identify different types of subclusters.

Machine learning (ML) can be defined as a set of methods that can automatically detect patterns in data. Within the field of ML, two main paradigms can be considered: supervised and unsupervised learning. The main difference is that supervised learning uses labelled data to classify labels, while unsupervised does not.

In relation to unsupervised learning, one of the most common tasks is clustering, which is a data mining technique for grouping unlabelled data based on their similarities or differences. In the literature, many different clustering algorithms have been proposed [9], [10]. However, despite such a diversity, some algorithms are very common and are widely used for very different tasks, such as K-Means or EM [11]. One of these popular algorithms is the probabilistic method Expectation-Maximization (EM) [12], which stands out for its simplicity and ease of implementation. EM algorithm guarantees to converge to a local maximum, which could not be the same as the global one. For this reason, it should be necessary to repeat the whole algorithm several times with different initial parameters (specially, the seed). In this research, we use only the Weka`s default seed. This aspect, which has not been considered in this research, will be analysed in detail in future works.

The 50 indicators used for the clustering process (and afterwards for describing the obtained clusters) are listed in Table I. They were selected from more than one hundred indicators provided by the HBS after recurrent evaluation of previous clustering results. They are grouped in 6 different classes, related to different features of energy poverty vulnerable groups.

TABLE I. INDICATORS USED FOR THE DESCRIPTION OF THE CLUSTERS CLASSIFIED BY CLASS. SOURCE: COMPILED BY THE AUTHORS USING HBS [5]

Class of indicators	Indicators included
General information	Household sequential number, Province capital, Size of the municipality, Population density, main breadwinner sequential number, Elevation factor of the household
Household characteristics - composition	Equivalent size of the household, Number of members, Size of the household, Number of members under 18, Number of members between 65 and 84, Number of members above 85, Number of students, Number of non-students, Type of households
Household characteristics - activity	Number of active members, Number of non-active members, Number of employed members, Number of unemployed members, Situation of the household regarding employment, Situation of the household regarding activity
Main breadwinner - Sociodemographic features	Age, Sex, Country of birth, Nationality (Spanish, foreign or both), Region of the nationality (for non-Spaniards), Level of education
Main breadwinner - Employment situation	Employment situation the previous week, Situation regarding activity, Was the main breadwinner employed the previous week?, Type of working day, Have you ever worked before?, Previous or current employment, Type of activity of employing company, Professional situation, Activity sector, Existence of employment contract, Type of employment contract, Socioeconomic situation
Features of housing stock	Tenure status, Type of building, Type of housing area, Type of house, Number of bedrooms, Year of construction, Inhabitable area, Hot water availability, Hot water energy source, Heating system availability, Heating systems energy source

C. *Studying the characteristics of each subcluster*

One of the main keys in this work is to be able to describe very clearly the different clusters that are formed. In order to solve this issue, it is proposed that clustering is followed by a stage in which a decision tree is inferred that allocates each instance to the cluster in which it belongs [13]. Thus, a set of five decision trees will be generated per group (G1-G3) in order to define a clear description of each cluster inside each group. Each decision tree is generated only with a subset of attributes that correspond to the family of attributes presented in Table 1. In a decision tree, each branch represents a choice between several values of a specific attribute. This way, all branches which predict a specific cluster can be used as an explicit description of that cluster.

The algorithm used in this work to generate the decision tree is the well-known C4.5 [14] (Weka`s implementation-J48), which, at each node of the tree, chooses the attribute of the data that most effectively splits its set of samples into subsets. The reason for using C4.5 is that the obtained models are easily interpreted, the algorithm is easy to implement, and it can deal with noise.

The rules of each decision tree were processed and grouped by clusters. This helped to identify the most relevant indicators, and which would differentiate and characterise each of the energy-poor clusters. In this sense, indicators that concentrated a larger number of households were considered more relevant in characterising a cluster.

Similarly, confusion matrices were used to identify those families of attributes for which two or more clusters could not be differentiated from each other. This made it possible to identify meeting points between different clusters, which provides information on how far apart these energy-poor subgroups might be from each other.

III. RESULTS

A. *Identification of energy poor households*

The use of HBS allows, as previously described, the division of the households of the Community of Madrid in six different groups which respond to different degrees of vulnerability towards energy poverty.

TABLE II. DESCRIPTION AND NUMBER OF HOUSEHOLDS AND PEOPLE IN EACH GROUP (REGARDING ENERGY POVERTY). SOURCE: COMPILED BY THE AUTHORS USING 2019 HBS [5]

Group		Household		People	
		Number	%	Number	%
G1	Monetary poverty and risk of energy poverty due to a high expenditure	152,684	5.8	380,295	5.8
G2	Monetary poverty and risk of energy poverty due to lack of adequate temperatures	217,714	8.3	721,762	10.9
G3	Risk of energy poverty due to a high expenditure	177,067	6.7	332,045	5.0
G4	Monetary and energy vulnerability	357,682	13.6	954,928	14.4
G5	Monetary vulnerability	351,471	13.3	989,056	15.0
G6	No vulnerability	1,381,569	52.4	3,233,687	48.9

Total	2,638,190	100	6,611,775	100

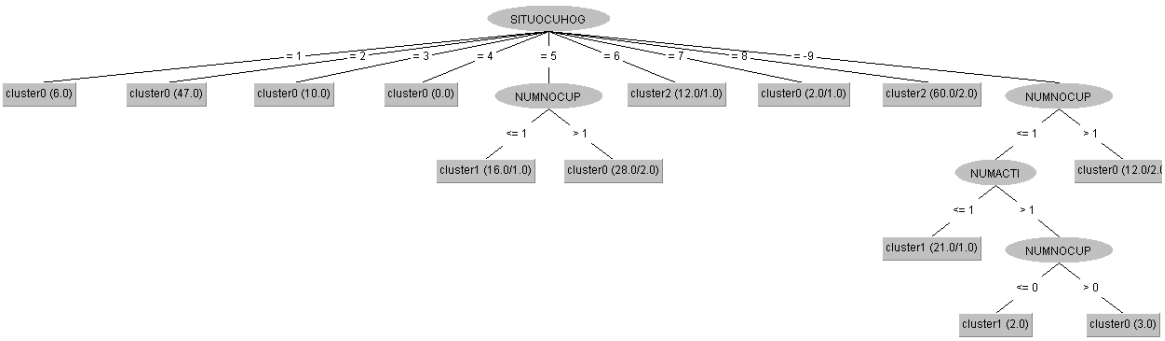


Fig. 1. Decision tree obtained for G2 in relation with the households' activity characteristics.

The number of households is calculated, as abovementioned, using the elevation factor which allows the sample (345 households) to represent the total number of households in the region (2,638,190). According to the results, 1 of every 5 households belong to the main three groups which are considered to represent or include the different types of energy poverty. This is a considerable amount of energy poor households to be analysed and described. Some of the differences between groups are clear, and they derived primarily from their relative situation regarding income and expenditure. Nevertheless, and since each group represents more than 150,000 households, unravelling the possible sub-clusters present in each of them may shed some light on features and differences within each group. This reinforces the pertinence of the clustering process, which results are presented next.

B. Clusters of energy poor households

In order to define the number of clusters per set of data (G1, G2 and G3), the algorithm EM was executed defining a specific number of clusters (from 2 to 5 clusters). Then, the number of clusters was chosen taking into account their corresponding likelihood. Thus, after executing the EM algorithm 4 times per set of data, the most efficient number of clusters were the following: G1: 5 clusters. G2: 3 clusters. G3: 4 clusters.

After defining the number of clusters, the algorithm C4.5 was applied to define the characteristics of each cluster. However, in this work, C4.5 is applied by using different subsets of attributes, so the description (decision tree) of the cluster would be done taking into account only some specific attributes at a time. The main indicators which were key to define the characteristics of the clusters are reflected in Table III.

The optimal subdivision of G1 yields 5 clusters. Out of these, 2 clusters correspond to households of 2 or more persons where the main breadwinner is active and whose tenancy regime is renting. Both clusters are primarily divided by the profile of the main breadwinner: in one case this person is a man over 50 years old and unemployed (n=12), and in the other it is either a man under 50 years old or a woman with a medium-high level of studies, who is employed (n=23).

Another 2 clusters of the G1 correspond to households where people are inactive and live in an owned dwelling with more than 25 years old and no mortgage. They are differentiated between single-person households, consisting of a female housewife living in a small dwelling (n=21), and households of 2 or more persons headed by a man who is retired and who live in a medium-sized dwelling (n=16). A final cluster, the smallest of them all (n=7), presents very varied characteristics for all groups of indicators and does not seem to fit into any particular profile.

TABLE III. KEY ATTRIBUTES OF EACH CLASS OF INDICATORS. SOURCE: COMPILED BY THE AUTHORS USING HBS [5]

Class of indicator	Key attributes
Household characteristics - composition	Number of members and members above 65
Household characteristics - activity	Number of active and unemployed members
Main breadwinner - Sociodemographic features	Age and sex of main breadwinner
Main breadwinner - Employment situation	Was the main breadwinner employed the previous week?
Features of housing stock	Tenure status, type of building, type of house, year of construction, inhabitable area

Regarding G2, the results show 3 well differentiated clusters for almost all indicator categories. Two clusters are associated with households where the main breadwinner is active, while the third is linked to households where the main breadwinner is retired or engaged in housekeeping.

As for households with active persons in G2, all of them share that at least one person is employed, and that the main breadwinner is less than 55 years old. Households of 3 or more persons are clustered separately, where a majority have children and live in a home owned with a mortgage; however, there is also a minority within this cluster with a home owned without a mortgage (n=106). On the other hand, a single cluster is formed by households of 2 or less persons with a low level of employment, where the housing characteristics and tenure regime are variable (n=39).

The third G2 cluster, made up of inactive main breadwinner's households (n=74), presents a quite diverse situation in terms of size and composition. This cluster shares both the tenure regime, corresponding to home ownership without mortgage, and other characteristics of the

dwelling, such as belonging to a multi-family building of 10 dwellings or more with more than 25 years old.

Finally, G3 has been differentiated into 4 clusters, two associated with active households and two with inactive households. Regarding the first two clusters, it is worth noting that in both there is at least one employed person. The size of these households is variable, as is their composition, although there is no person over 65 years old. It is worth noting that these two clusters differ fundamentally in terms of housing characteristics and tenure, with some households living in the smaller, rented dwellings (n=26), while others live in the larger, mortgaged dwellings (n=6).

Similar is the case for the two clusters of inactive households in G3. Both clusters share a large number of attributes, being indistinguishable in terms of their composition, socio-demographic or activity variables. They are also very similar in terms of tenure status, which in both cases corresponds to ownership without mortgage. The key parameters dividing the two clusters are, once again, housing characteristics: while one cluster groups households living in the smallest dwellings (n=39), there is an important group that resides mostly in single-family dwellings (n=21).

It is clear that the indicators related to the level of activity in the household, and those linked with the activity of the main breadwinner, are determinants for describing the clusters of the energy poverty groups. In all of them, pensioner households are clearly distinguished from active households and, among the latter, those with at least one employed person are differentiated from those without. This has been particularly remarkable in G1 and G2, which decision trees were able to correctly describe the clustering of households in 83.5% (main breadwinner activity) and 95.4% (household activity) of the cases, respectively.

Indicators associated with housing characteristics and tenure status also proved to be particularly relevant in describing these clusters. In the case of G3, it was this family of indicators that most accurately described the clusters, with 82.6% of households correctly described. The differentiation focused mainly on tenure status and household size, and less on other details such as the availability of heating equipment or the fuels used for heating.

Finally, with respect to the families of indicators associated with household composition and the sociodemographic characteristics of the main breadwinner, the indicators most used to differentiate the clusters were those referring to the number of persons in the household, their age and, in some cases, the sex of the main breadwinner.

IV. Discussion

In this paper we have considered the use of clustering algorithms and decision trees to identify and describe several subgroups of energy-poor households. One of the most noteworthy results is that, compared to the generation of a single tree based on all the available indicators, the pre-selection and grouping of indicators by classes for the subsequent generation of a variety of thematic decision trees has proved to be very useful. This approach has not only allowed us to interpret and characterise each cluster from different perspectives, but it has also helped to identify the meeting points between different clusters as well as their most significant differences. This could definitely help tailor energy poverty alleviation strategies to the needs of each subgroup of vulnerable households, and it highlights the relevance of domain knowledge when working on the identification of previously unknown clusters.

It should also be noted that the fact that the decision trees use certain indicators should not be interpreted as implying that these are the most relevant indicators for describing a cluster. In fact, the relevance of these indicators is related to their ability to differentiate between clusters, but it is not related to how important they are in determining whether a household is in energy poverty or not. Thus, although indicators such as the availability of heating systems in the dwelling [2] or the sex of the main breadwinner [15] have not been practically used by decision trees, this does not mean that these variables are no longer relevant; it represents that certain households (e.g. those without heating systems) are relatively homogeneously distributed among the clusters of a given energy poverty group. In any case, the overlap of both analyses could have a high potential for defining tailor-made solutions for the fuel poor, as well as for helping to improve policy interventions for vulnerable households.

Among the limitations of this study, perhaps the most remarkable is related to the level of data disaggregation. The figures represent the Community of Madrid as a whole, which is a heterogeneous region where very different realities coexist, and which could be affecting the clustering of vulnerable households. In this sense, although it would be convenient to carry out this type of analysis at lower geographic scales, the database used in this study (HBS) makes this level of detail more difficult to undertake. First, because this database does not distinguish between municipalities, but only differentiates them by size. And secondly, because the HBS does not collect data from all households, but only from a representative sample. Thus, filtering the database by the size of the municipalities could reduce the count of available records very significantly, which could limit the possibilities of applying the clustering and analysis techniques used in this study.

V. Conclusions

This study explores the use of machine learning techniques for the clustering of energy poor households. First, the EM algorithm was used on a battery of 50 HBS indicators to subdivide each of the three groups of households in energy poverty previously identified. Decision trees were then used to describe each of these clusters. This second part of the analysis was developed iteratively on a subselection of indicators, building 5 subgroups of thematic indicators from the 50 originally used, so as to describe each cluster in relation to a specific dimension (e.g. household composition, household employability status, housing characteristics). This approach has proven useful in identifying differences and similarities between clusters, as well as in determining common characteristics within each cluster.

From this clustering exercise, 12 subgroups of households in energy poverty have emerged. These clusters have proved useful in identifying common characteristics for each of these subsets of households, as well as for determining those that differentiate them from other subsets. In this sense, the employment situation was a determining factor in most cases when describing these clusters. In many

cases there were significant differences between these clusters in terms of housing characteristics.

This characterization of the subsets, throughout the several factors used, may allow to identify their different situations, which is a requisite for pointing out the needs of each of them regarding energy poverty alleviation. It is hoped that this type of analysis can help to refine housing intervention policies by making them more acute of the requirements of each subset.

The results of this study have been obtained from a database built on a sample of households and with a level of disaggregation limited to the regional scale. Although the results presented here reinforce the potential of this type of analysis, it would be necessary to conduct this analysis at lower geographic scales, closer to the municipal level, in order to describe more accurately the reality of households in a specific area. To this end, it would be advisable to increase the number of households considered in this type of databases, as well as to specifically invest in the creation of such databases by other entities from the public sector.

In any case, further experimentation would be needed to determine the number and type of clusters to be used in this domain, as well as to improve the interpretability of the results.

Acknowledgement

This research project was supported by the EPIU-Getafe Project (UIA04-212), co-financed by the European Regional Development Fund. This project was also supported by the Spanish Ministry of Science, Innovation and Universities (RTI2018-096036-B-C22), and by Universidad Carlos III (PEAVAUTO-CM-UC3M).

References

[1] B. Boardman, *Fuel Poverty: from Cold Homes to Affordable Warmth*. London, 1991.

[2] C. Sánchez-Guevara Sánchez, A. Sanz Fernández, M. Núñez Peiró, and G. Gómez Muñoz, "Energy poverty in Madrid: Data exploitation at the city and district level," *Energy Policy*, vol. 144, no. March 2019, 2020.

[3] M. (Ed. . Craglia *et al.*, "Artificial Intelligence: A European Perspective," Publications Office of the European Union, Luxembourg, 2018.

[4] Sage, "Artificial Intelligence in 2019: A handbook for business leaders," p. 22, 2019.

[5] Instituto Nacional de Estadística, "Encuesta de Presupuestos Familiares." 2019.

[6] C. Sánchez-Guevara Sánchez, A. Sanz Fernández, A. Hernández Aja, C. Sánchez-Guevara, A. Sanz Fernández, and A. Hernández Aja, "Income, energy expenditure and housing in Madrid: retrofitting policy implications," *Build. Res. Inf.*, vol. 43, no. 6, pp. 737–749, Nov. 2015.

[7] Government of Spain, "ERESEE 2020. Update of the long-term Strategy for Energy Rehabilitation in the Building Sector in Spain.," 2020.

[8] Ministerio para la Transición Ecológica, "Estrategia nacional contra la pobreza energética 2019-2024," 2019.

[9] R. Xu and D. Wunsch, "Survey of Clustering Algorithms," *IEEE Trans. Neural Networks*, vol. 16, no. 3, pp. 645–678, 2005.

[10] A. Saxena *et al.*, "A review of clustering techniques and developments," *Neurocomputing*, vol. 267, pp. 664–681, 2017.

[11] Y. G. Jung, M. S. Kang, and J. Heo, "Clustering performance comparison using K-means and expectation maximization algorithms," *Biotechnol. Biotechnol. Equip.*, vol. 28, no. sup1, pp. S44–S48, Nov. 2014.

[12] A. P. Dempster, N. M. Laird, and D. B. Rubin, *Maximum Likelihood from Incomplete Data Via the EM Algorithm* , vol. 39, no. 1. 1977.

[13] I. H. Witten, F. Eibe, M. A. Hall, and C. J. Pal, *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, 2017.

[14] R. Quinlan, *C4.5: Programs for Machine Learning*. Morgan Kaufmann, 2014.

[15] C. Sánchez-Guevara Sánchez, A. Sanz Fernández, and M. Núñez Peiró, "Feminisation of energy poverty in the city of Madrid," *Energy Build.*, vol. 223, p. 110157, 2020.

XIX Conferencia de la Asociación Española para la Inteligencia Artificial

I Taller de Inteligencia Artificial Cuántica (QUARTI)



Implementable Simple Hybrid Quantum Genetic Algorithm

1st Mikel Garcia de Andoin
Department of Electricity and Electronics
University of the Basque Country UPV/EHU
Bilbao, Spain
mikelgda@gmail.com

2nd Javier Echanobe
Department of Electricity and Electronics
University of the Basque Country UPV/EHU
Bilbao, Spain
franciscojavier.echanove@ehu.es

***Abstract*—Quantum machine learning (QML) is a relatively recent field of research in which the areas of Quantum Computing (QC) and Machine Learning (ML) are merged in different ways and at different levels. In this paper, we propose a quantum genetic algorithm (QGA) that is a direct translation of the classical genetic algorithm (GA). Compared to other existing works, our proposal allows a simpler and more direct implementation, and therefore with less gates. QGA is compared with its classical counterpart through a function optimization benchmark, showing that both algorithms are equivalent in terms of obtaining the optimal solution and the number of iterations needed. The results suggest future work of exploring similar algorithms and strategies in the search of quantum advantages.**

***Index Terms*—Quantum Computing, Quantum Algorithms, Genetic Algorithms, Optimization.**

I. Introduction

The theoretical and experimental scientific developments of the last decades have made it possible to advance and consolidate the discipline known as quantum computing (QC) [1]. This paradigm takes advantage of the particular laws of the quantum world such as superposition and entanglement, to perform calculations more quickly and efficiently. Although at the moment the systems developed are on a very small scale and are almost always limited to laboratory systems, it has been shown that for certain types of problems such as number factoring or search problems, quantum computers would have up to a exponential speed-up compared to conventional computers. For this reason, there are currently numerous research groups working hard to make quantum computers a reality in the near future.

The paradigm known as machine learning (ML) [2], encompasses a series of techniques and algorithms aimed at providing computers with prediction, generalization, and decision-making capabilities based on knowledge extracted from collections of reference data or learning data. This field has also experienced strong development in recent years due to the increasing capacity of computers as well as the availability of huge amounts of data to be processed. Among the most representative ML algorithms we can mention clustering, decision trees, support vector machines (SVM), Bayesian networks, neural networks or genetic algorithms. All these techniques have demonstrated their potential in many different applications, offering truly impressive results on numerous occasions. Today, ML techniques are at the heart of many so-called artificial intelligence systems. However, all these techniques require a large amount of hardware resources and have high computational complexity. Therefore, new techniques or methods to speed up its execution will be always of great value and interest.

In an effort to obtain joint benefit from ML and QC techniques, the area of research known as Quantum Machine Learning (QML) [3] has more recently emerged. This discipline aims to merge the two aforementioned paradigms to obtain techniques that allow to address different problems more efficiently. It can be found in the literature different ways in which that merge or fusion can be carried out. For example, classical ML algorithms can be applied to address problems in the field of QM [4], [5]. Also, classical ML algorithms can be implemented in quantum computers to speed-up its execution. See for example [6] for a review of the field. The work proposed here, falls in this second type; that is, we propose a quantum genetic algorithm (QGA) that can be truly implemented in a quantum computer in a very simple and direct way. In Section II we will make a review of existing quantum genetic algorithms and explain how our proposal has in comparison, interesting advantages.

The rest of the paper is organised as follows. Section II describe some of the main existing works dealing with the QGA topic. Section III explain the proposed QGA. In Section IV we present some experiments that show the feasibility of the proposal. Finally, section V is devoted to concluding remarks.

II. Related Works

Quantum genetic algorithms can be divided into two main groups, depending on the strategy used to test the fitness of the solution. On the one hand, algorithms based on different versions of the Grover's algorithm [7], use an oracle (sometimes called black-box) to inquire the state about its fitness. These algorithms are based on quantum circuits; i.e., they are implemented by means of basic quantum logic gates (e.g. Hadamard, Toffoli, Fredkin, CNOT). Examples of recent work for this kind of algorithms can be found in [8]–[10]. In these algorithms, an oracle $\hat{Q}$ takes the quantum state $|x\rangle$ as an

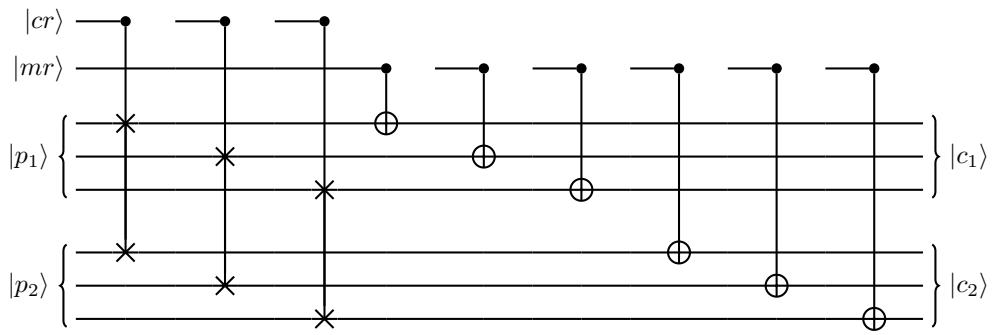


Fig. 1. Circuit for the simple QGA that implements the quantum crossover and mutation for a problem size n=3. Here, $p_{1,2}$ are the parents, and $c_{1,2}$ the children solutions. Each time the crossover or the mutation qubits are used, they are reset back to the state $|cr\rangle$ or $|mr\rangle$ respectively.

input, and without altering it, it deposit the fitness value in some ancilla qubits

$$\hat{Q}\,|x\rangle\,|0\rangle = |x\rangle\,|F(x)\rangle\,, \tag{1}$$

where $F(x)$ is expressed as its binary expansion, $F(x) = \sum_{k=0}^{\ell} 2^{\ell} F(x)_{\ell}$. This way, one could access the information about the evolution of the fitness function at each iteration while not destroying the quantum state. At the end, the measurement of the state yields the solution to the problem. In general, constructing such oracle for a given fitness function is non trivial, and it takes up to $2^{n-1}(2^n - 1)$ gates. That is, the complexity of the circuit (i.e. number of gates) grows exponentially [11].

On the other hand, there is the quantum annealing paradigm [12]. In this case, the system is initialized with a Hamiltonian (H_0) with an easily preparable ground state. The information of our optimization problem is encoded into the physical system by constructing an artificial Hamiltonian (H_1), so its ground state is the solution of the problem. Then, the system is adiabatically evolved towards the problem Hamiltonian for a time T,

$$H(t) = H_0(t) + H_1(t),\ H(0) = H_0,\ H(T) = H_1. \tag{2}$$

According to the adiabatic theorem, a physical system remains in the ground state if a perturbation acts on it slowly enough and as long as there is a sufficient energy gap between the ground and the first excited state. At the end of the evolution, the state is measured, with high probability of being in the ground state of the system, thus, the solution of the problem. The main drawback of this algorithm is that the construction of a suitable Hamiltonian can be a non-trivial task. Sometimes it may not be implementable on an adiabatic computer or annealer (e.g. D-Wave) or it is required to apply separation techniques (e.g. 'qbsolv'). Moreover, the Hamiltonian can be somewhat funny (e.g. Hamiltonians with terms with different energy scales), so that it is necessary to apply Shortcuts To Adiabacity techniques or Counter Diabatic terms [13].

In this paper we propose a QGA, implemented as quantum circuit, that does not precise an oracle. In consequence, the computation scheme is much simpler and therefore, its implementation is easy and straightforward, requiring much less hardware resources. In the next section we carefully explain the details of this algorithm showing how it is able to converge to a solution despite not using an oracle.

III. Proposed Algorithm

Let us describe in this Section how is the Quantum Genetic algorithm proposed in this work.

The codification of the solutions into a quantum state in the computational basis is straightforward. A classical bit corresponding to 0(1) is coded in the ground(excited) state of a qubit $|0\rangle(|1\rangle)$. This way, any solution can be generated by starting from the ground state and applying a Pauli-x gate ($|0\rangle = X\,|1\rangle\,,\ |1\rangle = X\,|0\rangle$) to every qubit that has a 1 in the corresponding position.

Our algorithm is based on a simple Genetic Algorithm schema, like the ones presented in [14], [15]. It starts by generating m random solutions of size n. The algorithm evolves towards the optimal solution iteratively, generating new solutions until a stop condition is met. Each iteration can be divided into 3 main steps: parent selection, crossover and mutation. The "hybrid" label of our algorithm comes from the fact that some of the steps are purely classical, in particular the parent selection and individual evaluation.

Parent selection: At the start of each iteration 2 different parents are selected depending on its fitness. The algorithm favours the selection higher fitness solutions, so that the better individuals advance in the evolution process. For this task we can use various strategies, from which we decided to use the traditional Roulette Wheel selection. This strategy assigns to each individual (x_i) a probability of being chosen (p_i) that is proportional to its relative fitness, $p_i = F(x_i)/\sum_{j=1}^{m} F(x_j)$ with $F(x)$ being the fitness function.

Crossover: In a classical GA, this step combines the bits of the parents into two new solutions. First it is decided whether or not the crossover happens, depending on the crossover rate $cr \in [0, 1]$. Then, using the single-point crossover, a

random point on the string is selected. The two children will be the combination of the opposite halfs of the parents. In our quantum algorithm, this crossover could be applied just relabeling the qubits.

We propose to use a more quantum approach by implementing a quantum version of the uniform crossover. We can use Fredkin (or controlled SWAP) gates acting on the corresponding qubits of both parents (Fig. 1). The control qubit contains the information of cr, such that the qubits of the parents are flipped with probability cr, $|cr\rangle = \sqrt{1-cr}\,|0\rangle + \sqrt{cr}\,|1\rangle$. This state can be generated by applying a rotation on the y axis of an angle of $\theta_{cr} = 2\arcsin(\sqrt{cr})$, $|cr\rangle = R_y(\theta_{cr})\,|0\rangle$.

Mutation: In the classical mutation step, each bit of the children can flip with a probability $mr \in [0, 1]$. This can be directly implemented in a quantum algorithm using CNOT gates. Similar to the previous step, this probability can be coded into a qubit that acts as the control of the CNOT gates acting on the children's qubits.

The crossover and mutation steps are implemented using a quantum circuit (Fig. 1). At the end of each circuit run, the children qubits are measured in the z axis. Once the new solutions are converted to bits, we can compute and assign the fitness to each one.

A new generation is completed after a new population is full, that's it, after $m/2 - 1$ iterations. To prevent the best individuals from being lost, we use replacement-with-elitism. The 2 individuals with the best fitness value are automatically introduced in the next generation. This assures that the solution that the algorithm returns is the best solution found so far.

The GA produces new generations until it meets the stopping condition. The most naive condition is to preset a maximum number of iterations, $maxIter$. We can also set a certain number of iterations to let the GA to evolve, $converIter$. If after $converIter$ generations no new best solution is found, the algorithm finishes.

The No Free Lunch Theorem [16] states that we can not find the set of parameters that optimizes an algorithm for every possible problem. In other words, there is no universal optimization strategy. The input parameters used to run GA have to be chosen by intuition or by running similar problems, i.e. empirically.

The proposed algorithm (Alg. 1) can be run on quantum computers without the need to use quantum error correction. Flip-errors can be absorbed as an extra probability to the mutation rate. Gate errors, in both Fredkin and CNOT gates, reduce the effective cr and mr. Dephasing noise does not interfere with our algorithm, as all the information is coded into the probabilities to measure the excited and the ground state. The only noise that could affect the outcome of the algorithm is the decoherence. Fortunately, the number of gates, and therefore the time the quantum state must be preserved, is linear with the size of the individuals $\mathcal{O}(n)$. More precisely, the qubit initialization is made with one single qubit gate per qubit. The number of Fredkin gates needed in the circuit is n, and each Fradkin gate can be implemented with 7 two-qubit gates [17]. For the mutation step the number of CNOT gates is two times the number of qubits per individual. Adding up all the gates for each step we clearly see that the number of gates in each circuit run is linear.

Algorithm 1 Simple Quantum GA

Require: Fitness function F, crossover rate cr, mutation rate mr, generation size m, stopping condition
1: Generate m random solutions
2: **repeat**
3: **for** $i = 1 : m/2 - 1$ **do**
4: Select 2 parents
5: Run the quantum circuit to generate 2 children (see Alg. 2)
6: Evaluate the children fitness value
7: **end for**
8: Merge the children and the 2 best individuals into the next generation
9: **until** stopping condition is met
10: **return** The best individual

Algorithm 2 Quantum circuit description

Require: Parents $p_{1,2}$, crossover rate cr, mutation rate mr
1: $N =$ bit length of parent
2: $\theta_{cr} = 2\arcsin(\sqrt{cr})$
3: Initialize quantum state, $|0\rangle^{\otimes 2N+2}$
4: Apply $R_y(2\arcsin(\sqrt{cr}))$ gate to the crossover qubit
5: Apply $R_y(2\arcsin(\sqrt{mr}))$ gate to the mutation qubit
6: **for** all bits of both parents **do**
7: **if** bit j of parent i == 1 **then**
8: Apply X gate to qubit j of parent i
9: **end if**
10: **end for**
11: **for** $j = 1 : N$ **do**
12: Apply Fredkin to qubit j of both parents as the targets with the crossover qubit as the control
13: Reset the crossover qubit and apply $R_y(2\arcsin(\sqrt{cr}))$
14: **end for**
15: **for** all qubits of both parents **do**
16: Apply CNOT gate to the parent qubit with mutation qubit as the control
17: Reset the mutation qubit and apply $R_y(2\arcsin(\sqrt{mr}))$
18: **end for**
19: **return** The measure of parent qubits in the z axis

IV. Experiments

To test the proposed algorithm we have realized some simulation experiments over commonly used benchmarks. More precisely, the problem to solve is to find the $x = [x_1, x_2, \ldots, x_i, \ldots, x_n]$, with x_i a binary variable with values

$x_i = \{0, 1\}$, that minimizes a given function $F(x) \in \mathbb{R}$. We have used the same 5 benchmarking functions as in [18]:

$$F_1(x) = \sum_{i=1}^{n} x_i^2, \ |x_i| < 5.12, \ \min(F_1) = 0 \text{ for } x_i = 0, \quad (3)$$

$$F_2(x) = 20 + e - 20 \exp\left(-\frac{1}{5}\sqrt{\frac{\sum x_i^2}{n}}\right) - \exp\left(\frac{\sum \cos(2\pi x_i)}{n}\right), \quad |x_i| < 30, \ \min(F_2) = 0 \text{ for } x_i = 0, \quad (4)$$

$$F_3(x) = 10n + \sum_{i=1}^{n} \left(x_i^2 - 10\cos(2\pi x_i)\right), \quad |x_i| < 5.12, \ \min(F_3) = 0 \text{ for } x_i = 0, \quad (5)$$

$$F_4(x) = \sum_{i=1}^{n} x_i^2 + \left(\sum_{i=1}^{n} \frac{x_i}{2}\right)^2 + \left(\sum_{i=1}^{n} \frac{x_i}{2}\right)^4, \quad -5 \leq x_i \leq 10, \ \min(F_4) = 0 \text{ for } x_i = 0, \quad (6)$$

$$F_5(x) = \sum_{i=1}^{n} i x_i^2, \ |x_i| < 5.12, \ \min(F_5) = 0 \text{ for } x_i = 0, \quad (7)$$

where n is the dimension of the problem.

For the coding, we have used a Gray coding, commonly used in other GA [19, Sec 8.2]. The fitness is calculated using the all-vs-all simple sampling method [19, Sec 15.2.1]. To obtain the fitness of the initial population the individuals are compared with themselves. As the simulation of a quantum circuit has a limit on the number of qubits we can work with, we have used 30 bit solutions with dimension 3 problems. That is, we have coded each number with 10 bits.

For running the classical GA we used *Matlab R2021a*. In order to have a fair comparison, we used two versions of GA, one with the uniform and the other with the single-point crossover strategy. We repeated the algorithm 100 times, taking the mean value.

The QGA has been simulated using the *Matrix Product State* simulator from IBM, via the IBM Quantum Lab interface [20]. As this is an ideal simulator, there is no restriction on the topology of the circuit. However, we are limited to the open access services. This limits the response times of the queries to process the simulations. To avoid overflowing the system, we have only run QGA three times per benchmark function, with a maximum of $3 \times 5 \times 50 \times 9 = 6750$ circuit runs.

In order to compare the 3 versions of the algorithm, we have used the same input parameters in all of them (Tab. I). The results of the benchmarking are presented on the table II. If we compare the results we obtained with QGA to the mean values of those of GA, we see that the performance of the two is similar.

TABLE I
INPUT PARAMETERS

Length of individual (n)	30
Population size (m)	20
Crossover rate (cr)	0.7
Mutation rate (mr)	0.01
$maxIter$	50
$converIter$	10

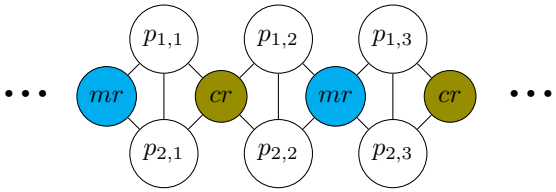


Fig. 2. Proposed circuit topology and qubit distribution for QGA, with being the qubits the vertex and the controlable couplings the edges. Here, each parent qubit $p_{i,j}$ is connected such way we can implement the circuit in Fig. 1 without using SWAP gates, at the cost of adding $n - 2$ extra qubits.

If we wanted to run QGA in a real quantum computer and optimize the implementation, we have to take into account the topology of the circuit. In the proposed algorithm, ideally we only need 1 qubit to code the mutation and the crossover rate. However, this qubit has to be connected with all other qubits at the same time. Due to their topology, this is far from doable in the current and near-term quantum computers. One solution to this problem is to use SWAP gates to move the qubit around the circuit. Although this is possible, each time a SWAP gate is used there is an increasing probability that the information on the qubit gets destroyed due to gate errors. To avoid this, we can instead duplicate the qubit in different positions on the circuit. This would minimize the information loss by using more qubits than the required. An example of this is shown in Fig. 2, where we have a topology of the circuit that corresponds to a plane graph. This feature makes it suitable to be built with the current superconducting circuits technology.

V. Conclusions and Future Perspectives

In this paper, we present a simple algorithm that implements a quantum version of GA. The proposed QGA directly translates the crossover and mutation strategies into a quantum circuit. This algorithm does not require any oracle, leading thus to a more simple and direct implementation. Through a benchmarking of QGA against the classical GA, we are confident that both approaches are equivalent in terms of obtaining the optimal solution of a problem, up to a reasonable uncertainty. A formal proof for this statement is beyond the scope of this work.

The quantum version of GA we proposed here does not suppose any speedup in the computational complexity. However, it is shown that the algorithm can be implemented efficiently in a quantum computer. This could open the door to develop a quantum version of other heuristic or metaheuristic algorithms based on the coding of classical probabilities directly as the

TABLE II
RESULTS OF THE BENCHMARKING

Functions	QGA			GA uniform	GA single-point
	Val 1	Val 2	Val 3	Mean	Mean
F1	0.0019	0.0013	0.0015	0.0066	0.0094
F2	3.6	6.8	1.4	2.9	3.3
F3	17	8.6	6.7	4.1	3.4
F4	0.0011	0.00064	0.0020	0.011	0.042
F5	0.0022	0.27	0.61	0.017	0.024

probability of measuring a quantum state in a certain basis. This could lead to the finding of new strategies that truly take advantage of the quantum behaviour of the system, such as superposition of states, entanglement or quantum parallelization.

REFERENCES

[1] National Academies of Sciences, Engineering, and Medicine, "Quantum Computing: Progress and Prospects," Washington, DC: The National Academies Press, 2019.

[2] C. M. Bishop, "Pattern Recognition and Machine Learning," Springer, 2006.

[3] M. A. Nielsen, I. L. Chuang, (2010), "Quantum Computation and Quantum Information (2nd ed.)," Cambridge, Cambridge University Press, 2010.

[4] D. G. Cory, N. Wiebe, C. Ferrie, C. E. Granade,"Robust Online Hamiltonian Learning", in New J. Phys., Vol. 14 (10), 2012.

[5] M. Krenn, "Automated Search for new Quantum Experiments," Physical Review Letters", Vol. 116 (9), 2016.

[6] V. Dunjko, H.J. Briegel "Machine learning & artificial intelligence in the quantum domain: a review of recent progress," Reports on Progress in Physics, vol. 81(7), 2018.

[7] L. K. Grover, "A fast quantum mechanical algorithm for database search," in Proc. 28th An. ACM STOC, pp. 212-219, 1996.

[8] M. Udrescu, L. Prodan, and M. Vlăduţiu, "Implementing quantum genetic algorithms: a solution based on Grover's algorithm," in Proc. 3th CF, pp. 71-82, 2006.

[9] A. Malossini, E. Blanzieri, and T. Calarco, "Quantum Genetic Optimization," in IEEE Trans. Evol. Comput., vol. 12, no. 2, pp. 231-241, 2008.

[10] R. Ibarrondo, "Quantum Genetic Algorithms: towards the design of evolutionary algorithms in a quantum computer," in ADDI, 2020.

[11] C. K. Li, R. Roberts, and X. Yin, "Decomposition of unitary matrices and quantum gates," in Int. J. Quant. Inf., vol. 11, no. 1, pp. 1350015, 2013.

[12] W. Shu, and B. He, "A Quantum Genetic Simulated Annealing Algorithm for Task Scheduling," in ISICA 2007: Adv. Comput. Int., vol. 4683, pp. 169-176, 2007.

[13] N. N. Hegade, K. Paul, et al, "Shortcuts to Adiabaticity in Digitalized Adiabatic Quantum Computing," in Phys. Rev. Appl., vol. 15, pp. 024038, 2021.

[14] D. E. Golberg, "Genetic Algorithms in Search, Optimization and Machine Learning," MA: Addison-Wesley, 1989.

[15] J. McHall, "Genetic algorithms for modelling and optimisation," J. Comput. Appl. Math., vol. 184, pp. 205-222, 2005.

[16] Y. C. Ho, and D. L. Pepyne, "Simple Explanation of the No-Free-Lunch Theorem and Its Implications," J. Optim. Theory Appl., vol. 1, pp. 549-570, 2002.

[17] N. Yu, and M. Ying, "Optimal simulation of three-qubit gates," 2013. https://arxiv.org/abs/1301.3727v1

[18] S. P. Lim, and H. Haron, "Performance of Different Techniques Applied in Genetic Algorithm towards Benchmark Functions," In Proc. 5th ACIIDS, vol. 1, pp. 255-264, 2013.

[19] A. P. Engelbrecht, "Computational Intelligence: an introduction Second Edition," Wiley Publishing, 2007.

[20] IBM Quantum. https://quantum-computing.ibm.com/

Mutual Reinforcement between Neural Networks and Quantum Physics

1st Yue Ban
Department of Physical Chemistry
University of the Basque Country UPV/EHU
Bilbao, Spain
ybanxc@gmail.com

2nd Javier Echanobe
Department of Electricity and Electronics
University of the Basque Country UPV/EHU
Bilbao, Spain
franciscojavier.echanove@ehu.eus

3rd Erik Torrontegui
Departamento de Física
Universidad Carlos III de Madrid
Instituto de Física Fundamental IFF-CSIC
Madrid, Spain
eriktorrontegui@gmail.com

4th Jorge Casanova
Department of Physical Chemistry
University of the Basque Country UPV/EHU
IKERBASQUE, Basque Foundation for Science
Bilbao, Spain
jcasanovamar@gmail.com

***Abstract*—Quantum machine learning emerges from the symbiosis of quantum mechanics and machine learning. In particular, the latter gets displayed in quantum sciences as: (i) the use of classical machine learning as a tool applied to quantum physics problems, (ii) or the use of quantum resources such as superposition, entanglement, or quantum optimization protocols to enhance the performance of classification and regression tasks compare to their classical counterparts. This paper reviews examples in these two scenarios. On the one hand, a classical neural network is applied to design a new quantum sensing protocol. On the other hand, the design of a quantum neural network based on the dynamics of a quantum perceptron with the application of shortcuts to adiabaticity gives rise to a short operation time and robust performance. These examples demonstrate the mutual reinforcement of both neural networks and quantum physics.**

***Index Terms*—Neural Network, Quantum Sensing, Machine Learning, Quantum Perceptron**

I. Introduction

Artificial neural networks [1] are one of the most fruitful computational models of machine learning (ML) [2] thanks to the blooming of deep learning [3], [4] in the recent years. Inspired by biological brains, artificial neural networks (NNs) are organized in layers feeding signals into other neurons allowing parallel-processed [5] and universal [6] computing. Introduced first in 1943 as a computational model based on discrete threshold logic algorithms [1] and linked to biological learning theory [7], the neuron activation mechanism –i.e. perceptron, was characterized [8] and generalized to graded response activation potentials [5]. When nested with other perceptrons, the resulting multilayer network becomes universal with the capacity to approximate any continuous function [6], [9], based on "universal approximation theorem" claiming that any continuous function can be written as a linear combination of sigmoid functions.

The topology of NNs has increased in parallel with the computer hardware improvements [10]. As a result, the enlargement of the depth has promoted the ability of a NN to process the exponentially increasing amount of data [11] and complexity of algorithms [12] in the information explosion era [13]. Thus, the versatility of NNs covers diverse fields such as economy, industry, transportation, or science among others. More recently, the so-called deep learning paradigm and particularly the Convolutional Neural Networks (CNNs) [14], have shown incredible capabilities in applications such as speech [15] or object recognition [16], spam filters [17], vehicle control [18]. Neural networks can be trained to perform tasks without the programmer necessarily detailing how to do it. Due to this property, CNNs are also employed to develop the expected artificial intelligence.

Over the last two decades, quantum physics has experienced its second revolution giving rise to new quantum technologies. Thanks to quantum control, matter can be manipulated at the single particle level by exploiting quantum resources such as entanglement, superposition or squeezing of states in various platform registers with high fidelity [19]–[23]. All the progress indicates that quantum physics will offer outbreak for the future coming in a wide variety of forms, from quantum cryptography [24], quantum sensing [25], to quantum computing [26] among others. Some examples of technologies emerged in this second quantum revolution are atomic sensors providing unprecedented high resolution and efficiency in the detection of external fields [27]–[31], quantum channels [32] that make use of entanglement for unbreakable communications [33], or quantum computers that use superposition to execute parallel-processing computations performing specific tasks with higher efficiency than their classical counterparts [34].

The meet of ML and quantum physics gives birth to a novel field, quantum machine learning (QML) [35], [36], bringing a lot of progress on both fields. On the one hand, the universality of NNs allows to enhance the accuracy and efficiency of quantum protocols. ML is diversely useful in

measurements protocols [37]–[39], states preparation [40], [41], entanglement and states classification [42], [43], quantum communication [44], learning Hamiltonians [45] and handling with open quantum system dynamics [46]–[48]. Recently, ML also starts to attract its attention in quantum sensing and metrology, in particular, adaptive protocols for phase [49], [50] and parameter estimation [51], [52], or calibrating quantum sensors [53]. On the other hand, the use of quantum resources allows improvements of the NN accuracy in classification and regression tasks compared to their classical counterparts. Since the seminal publication by Kak [54], different efforts have been made to reproduce the nonlinear response for the perceptron at the quantum level and its nesting to the design of artificial quantum neural networks (QNNs) [55]–[63]. In particular, in Reference [61] it is already demonstrated that a QNN based on the quantum perceptron gates evolving adiabatically with the sigmoid-like activation has the ability to approximate continuous functions.

In this article, we review two examples of QML, demonstrating the uses of NNs in quantum physics. This article is structured as follows. In Sec. II, we review the use of a classical NN applied to quantum sensing [64]. Such a NN is used for parameter predictions of an external field, showing its performance in a wide range of working regimes, even beyond the rotating wave approximation (RWA) [65], [66] with a training/validation/test dataset fully obtained from experimental measurements. The establishment, training and operation of the NN require a minimal knowledge of the physical system in contrast with Bayesian inference methods [31]. In Sec. III, we review the quantum perceptron where the sigmoid-like activation is mimic by the excitation probability of a qubit. Implemented as an efficient and reversible unitary operation, the resulting QNN is encoded by an Ising model [61]. With the aid of shortcuts to adiabaticity (STA) [62] the perceptron can provide faster and more robust nonlinear responses. The results of these work indicate the mutual reinforcement of machine learning and quantum physics.

II. Application of NNs in quantum sensing

Target parameters have been usually encoded in harmonic responses of quantum sensors [65], [66]. However, this approach limits the working regime of the sensor, as the deviation from harmonic responses leads to a failure on the parameter estimation. In addition, being in this limited working regime requires enough pre-knowledge of the physical model. In this scenario, a well-trained feed-forward NN (such as the one proposed in [64]) can predict the parameters of the external field just from the experimental acquisitions without the necessity of learning the underlying physical model. In particular, in the work conducted in [64] we focused on an $^{171}\mathrm{Yb}^{+}$ atomic quantum sensor [27]–[31] and demonstrated that its working regime was extended to complex scenarios.

The atomic sensor works as follows: In the $^2S_{\frac{1}{2}}$ manifold of a $^{171}\mathrm{Yb}^{+}$ ion, there are four hyperfine levels $|0\rangle$, $|\acute{0}\rangle$, $|1\rangle$ and $|-1\rangle$, with a Zeeman splitting in the late three states induced by a static magnetic field B_z. Two microwave drivings with the same amplitude Ω and resonant with $|0\rangle \rightarrow |1\rangle$ and $|0\rangle \rightarrow |-1\rangle$ are applied to cancel potential noisy fluctuations in B_z. Via the detection of the state transition between the $|\acute{0}\rangle \rightarrow |1\rangle$ (or $|\acute{0}\rangle \rightarrow |-1\rangle$), a target electromagnetic field with frequency ω_{tg} and amplitude Ω_{tg}, in the form of $\Omega_{\mathrm{tg}}\cos(\omega_{\mathrm{tg}})$ can be probed based on the responses that the sensor gives. In the dressed state basis $\{|u\rangle, |d\rangle, |B\rangle, |D\rangle\}$, where $|u\rangle = (|B\rangle + |0\rangle)/\sqrt{2}$, $|d\rangle = (|B\rangle - |0\rangle)/\sqrt{2}$, $|D\rangle = (|-1\rangle - |1\rangle)/\sqrt{2}$, $|\acute{0}\rangle = |\acute{0}\rangle$, with $|B\rangle = (|-1\rangle + |1\rangle)/\sqrt{2}$, the state probability $P_D(t)$ is considered as the response given by the sensor with $P_D(0) = 1$. Under the condition that RWA is valid, $\Omega_{\mathrm{tg}} \ll \Omega \ll \omega_{\mathrm{tg}}$, the sensor releases the harmonic response. However, when the condition of RWA is not satisfied, i.e., with large Ω_{tg}, small ω_{tg} or small detuning ξ in $\omega_{\mathrm{tg}} = \omega_1 - \omega_{\acute{0}} + \xi$, the sensor offers complex responses that can be analized with a feed-forward NN.

We demonstrated our method in the following manner: although in reality the training/validation/test data are taken from experimental measurement, their acquisitions with/under the equivalent conditions can be numerically simulated. At every t_i for each measurement, the result is either 0 or 1. With the shot times $n \in 1, 2, ..., N_m$, the results become $P_i = \sum_{n=1}^{N_m} z_{n;i}/N_m$, where $z_{n;i}$ is from a Bernoulli distribution deriving from the theoretical probability $P_D(t_i)$. The NN has the input from the measured data $\mathbf{X} = \{P_1, P_2, ..., P_{N_p}\}$ and sends the outputs $\mathbf{Y} = \{y_1, y_2\}$ after processing, aiming at the targets $\mathbf{A} = \{a_1, a_2\} = \{\Omega_{\mathrm{tg}}, \xi\}$. The elements P_i in the input data set $\mathbf{X}$ are collected at every time instant $t = t_i$ with shot times N_m, in the time interval $[0.5t_0, t_0]$, where we choose the period for the ideal harmonic response as t_0.

In order to obtain the high accuracy for the approximation $F(\mathbf{X}) = \mathbf{Y} \approx \mathbf{A}$, we tune the parameters of the NN and adopt the set to give the high estimation precision in Table I. Levenberg-Marquardt backpropagation, one of the gradient descent algorithms, is applied to train the NN. To build a training/validation/test dataset, 241 examples in $\Omega_{\mathrm{tg}}/(2\pi) \in [1, 25]$ kHz multiplied by 11 examples chosen in $\xi/(2\pi) \in [-0.3, 0.3]$ kHz are extracted. Here, we set the above ranges for the parameters as one example to show the estimation abilities of the NN beyond the harmonic response, i.e., $\Omega_{\mathrm{tg}} = 2\pi \times 1$ kHz and $\xi = 0$ kHz. Of course, one can also change the above ranges for the estimation parameters. In addition, the precision

TABLE I

Number of neurons in the input layer N_p	101
Number of hidden layers	5
Number of neurons in each hidden layer	40 20 12 6 3
Activation function in each hidden layer	$y = \tanh(x)$
Activation function in the output layer	$y = x$
Learning rate	5×10^{-3}
Shot times for each expectation value	100
Repetition times for each simulated acquisition	20

Parameters for training the NN to predict $\mathbf{Y} = \{y_1, y_2\}$ approximating $\mathbf{A} = \{\Omega_{\mathrm{tg}}, \xi\}$. With the application of the above set, the average value of the accuracy is above 0.97.

of a NN is not sensitive to the choice of the activation function. In the work conducted in [64], we consider $N_m = 100$ shot times for each expectation value. The finite measurement times gives rise to shot noise, bringing statistical fluctuations. To this end, 20 times is repeated for each simulated acquisition. Therefore, the total training/validation/test dataset contains $241 \times 11 \times 20$ examples, 70%, 15%, 15% from which consist of the training, validation and test sets.

With the presence of shot noise, the estimation of $\Omega_{\rm tg}$ and ξ behave in different manners. While the accuracy of the estimation of $\Omega_{\rm tg}$ keeps high, it is more difficult for the NN to learn ξ at small $\Omega_{\rm tg}$. We divide the whole dataset into the following intervals: $\Omega_{\rm tg}/(2\pi) \in [1, 25]$ kHz, $[3.4, 25]$ kHz, $[8.2, 25]$ kHz, $[13, 25]$ kHz, $[17.8, 25]$ kHz and $[22.6, 25]$ kHz, but keeping the same range for $\xi/(2\pi) \in [-0.3, 0.3]$ kHz. For all the above ranges, the correlation coefficient R measures the linear dependence between the outputs of the NN and the targets. The linear regression of the outputs y as the function of the targets $y = \alpha a + \beta$ (with the targets a) is compatible with the perfectly matching linear relation $y = a$. When the NN provides better estimation, the outputs approach the targets, $R \to 1$, and $\alpha \to 1$, $\beta \to 0$. In Table II, NN training results illustrated by R, α and β are listed in different intervals of $\Omega_{\rm tg}/(2\pi)$ with $\xi/(2\pi) \in [-0.3, 0.3]$ kHz. From Table II, we

TABLE II

$\Omega_{\rm tg}/(2\pi)$ (kHz)	R	$1-\alpha$	β
$[1, 25]$	0.99356	10^{-2}	$6 \cdot 10^{-3}$
$[3.4, 25]$	0.99863	$4 \cdot 10^{-3}$	$3 \cdot 10^{-3}$
$[8.2, 25]$	0.99972	$7 \cdot 10^{-4}$	$3 \cdot 10^{-4}$
$[13, 25]$	0.99984	$3 \cdot 10^{-4}$	$2 \cdot 10^{-4}$
$[17.8, 25]$	0.99996	10^{-4}	$4 \cdot 10^{-5}$
$[22.6, 25]$	0.99999	$7 \cdot 10^{-5}$	$2 \cdot 10^{-5}$

NN training results illustrated by R, α and β in different intervals of $\Omega_{\rm tg}/(2\pi)$ with $\xi/(2\pi) \in [-0.3, 0.3]$ kHz. The correlation coefficient R measures the linear dependence between the outputs from the NN and the targets, while the linear regression of the outputs is expressed in the function of the targets $y = \alpha a + \beta$.

can see higher accuracy that the outputs approach the targets is obtained when $\Omega_{\rm tg}$ is larger. Near the harmonic response regime, i.e. $\Omega_{\rm tg} = 2\pi \times 1$ kHz, it is more difficult for the NN to learn different values of ξ, as the fluctuation of the responses caused by shot noise can even exceed the difference of the ideal sensor response obtained from the theoretical model. In fact, to distinguish responses from shot noise close to the harmonic regime is difficult, independently of the technique adopted.

One may choose the dataset with the range $\Omega_{\rm tg}/(2\pi) \in [8.2, 25]$ kHz and $\xi/(2\pi) \in [-0.3, 0.3]$ kHz, randomly pick up N examples with target parameters outside the dataset and check the outputs from the NN. The average value of the accuracy $F_i = \frac{1}{N}\sum_{j=1}^{N} |y_i^j - a_i^j|/a_i^j, (i = 1, 2)$ is above 0.97. Of course, one can detect responses at longer time or increase the shot times N_m to improve the detection. However, more energy and time cost with more experiment resources should be paid. To make a balance between the gains and the cost is always needed. We demonstrate In Ref. [64] that in a $^{171}Yb^+$ atomic sensor, adequately trained neural networks enable to estimate the parameters of the external field in regimes presenting complex responses under the shot noise due to a finite number of measurements.

III. Improved quantum perceptron with STA

In Ref. [61], a quantum perceptron labelled by the subscript j offering the nonlinear response to an input field is constructed via a unitary transformation, written as

$$\hat{U}_j(\hat{x}_j; f)|0_j\rangle = \sqrt{1 - f(\hat{x}_j)}|0_j\rangle + \sqrt{f(\hat{x}_j)}|1_j\rangle. \quad (1)$$

This transformation is motivated by the *resting* and *active* states of a classical neuron that in the quantum description are encoded by the ground $|0\rangle$ and excited $|1\rangle$ states of a qubit. The sigmoidal activation potential of the neuron is codified by the excitation probability of the qubit $f(\hat{x}_j) = |\langle 1_j|\hat{U}(\hat{x}_j; f)|0_j\rangle|^2$. In a feed-forward network, the perceptron gate is conditioned on the field generated by neurons in earlier layers, $\hat{x}_j = \sum_{i=1}^{k} w_{ji}\hat{\sigma}_i^z - b_j$, with similar weights w_{ji} and biases b_j as classical networks.

We construct this perceptron gate evolving a qubit with the Ising Hamiltonian

$$\begin{aligned} \hat{H} &= \frac{1}{2}\left[\hat{x}_j\hat{\sigma}_j^z + \Omega(t)\hat{\sigma}_j^x\right] \quad (2) \\ &= \frac{1}{2}\left[\sum_{i=1}^{k}(w_{ji}\hat{\sigma}_i^z\hat{\sigma}_j^z) - b_j\hat{\sigma}_j^z + \Omega(t)\hat{\sigma}_j^x\right]. \end{aligned}$$

The qubit is controlled by an external transverse field $\Omega(t)$, has a tuneable energy gap and interacts with other neurons through $\hat{x}_j$. The Hamiltonian Equation (2) has the reduced ground state

$$|\Phi(x_j/\Omega(t))\rangle = \sqrt{1 - f(x_j/\Omega(t))}|0\rangle + \sqrt{f(x_j/\Omega(t))}|1\rangle, \quad (3)$$

with a sigmoid-like excitation probability of the form

$$f(x) = \frac{1}{2}\left(1 + \frac{x}{\sqrt{1+x^2}}\right). \quad (4)$$

Originally, the final state (3) is achieved in three steps: (i) set the perceptron to the superposition $|+\rangle = \mathcal{H}|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ with a Hadamard gate; (ii) instantaneously boost the magnetic field $\Omega(0) = \Omega_0 \gg |\hat{x}_j|$; (iii) adiabatically ramp-down the transverse field $\Omega(t_f) = \Omega_f$ in a time t_f, to do the transformation $\mathcal{A}(\hat{x}_j)|+\rangle \simeq |\Phi(\hat{x}_j/\Omega_f)\rangle$. However, a faster dynamics is always desirable, as a shorter operation time to construct the perceptron response (4) can enhance the performance and reduce the decoherence in the quantum registers. Inverse engineering (IE) methods [67] can accelerate the dynamics providing different sigmoid-like excitation probabilities of the qubit leading to an smoother external driving profile $\Omega(t)$ that favors its experimental realization. Moreover, as universality does not depends on the specific shape of the sigmoid-like activation function [6], the resulting improved QNN is still universal [62].

More in detail, we parameterize the dynamical state with the undetermined polar θ and athimuzal β auxiliary angles on the Bloch sphere,

$$|\Psi(t)\rangle = \cos(\theta/2)e^{i\beta/2}|0\rangle + \sin(\theta/2)e^{-i\beta/2}|1\rangle. \quad (5)$$

Substituting this state $|\Psi(t)\rangle$ into the time-dependent Schrödinger equation driven by the Hamiltonian in Equation (2), we obtain that the auxiliary angles satisfy the following coupled differential equations,

$$\Omega(t) = \dot{\theta}/\sin\beta, \quad (6)$$
$$x_j = \dot{\theta}\cot\theta\cot\beta - \dot{\beta}. \quad (7)$$

Setting the wavefunction $|\Psi(0)\rangle = |+\rangle$ and $|\Psi(t_f)\rangle = |\Phi(x_j/\Omega_f)\rangle$ at the initial and final times imposes boundary conditions on $\theta(0)$ and $\theta(t_f)$. Moreover, Equation (6) also imposes similar boundary conditions on $\dot{\theta}(0)$ at and $\dot{\theta}(t_f)$ once the initial and final values of the external transverse field $\Omega(0)$ and $\Omega(t_f)$ are specified. The set of Eqs. (6) and (7) are solved by first, interpolating with a polynomial ansatz the function $\theta(t)$ so the boundary conditions of θ and $\dot{\theta}$ are fulfilled at $t = 0$ and t_f, subsequently, deriving $\beta(t)$ from Equation (7). Finally, once $\theta(t)$ and $\beta(t)$ are fully determined, $\Omega(t)$ is deduced from Equation (6), see [62] for more details. As a result, the initial state $|\Psi(0)\rangle$ is not necessarily the eigenstate of the Hamiltonian avoiding large $\Omega(0)$ values leading to a smooth and less experimentally demanding $\Omega(t)$ profile compared to fast-quasi-adiabatic implementations [61]. Moreover, the IE protocol allows a shorter operation time to construct a quantum perceptron and a stable performance with respect to timing errors and the variations of the input potential. In Ref. [62], by using STA we propose a speed-up quantum perceptron which has faster performance compared to fast-quasi-adiabatic protocols and enhanced robustness against imperfections in the controls. A deep QNN consisting of a number of our perceptrons can be implemented in physical platforms such as NV center in diamond, trapped ions, and superconducting circuits. One can find the training of such a QNN using gradient descent in the same way as Reference [61] where the example of searching prime numbers has been given using Python language with quantum perceptron gates evolving adiabatically. The neural potential of each perceptron can include multi-qubit interactions deviating from the current network paradigm of additive activations so that one can avoid internal hidden layers in a QNN without sacrificing approximative power for information processing tasks, see Ref [63].

IV. Conclusion

In this paper, we introduce the application of NNs on quantum sensing and the development of QNNs by using quantum resources. Particularly, we review the neural-network-based atomic quantum sensor of a $^{171}Yb^{+}$ ion, working in the regime of complex responses beyond harmonic ones. In general, the protocol is applicable to arbitrary quantum detection scenarios. We also review the construction of a quantum perceptron with IE strategies, an efficient method of quantum control, providing faster perceptron performance as well as enhanced robustness against imperfections in the controls. Decoherence during the physical implementation in quantum registers can be reduced due to the short operation time of the quantum perceptrons embedded in a QNN. We hope that more complex quantum problems with different environmental noise will be addressed by NNs, and the performance and training of NNs will be improved by quantum resources.

Acknowledgment

We acknowledge financial support from Spanish Government via PGC2018-095113-B-I00 (MCIU/AEI/FEDER, UE), Basque Government via IT986-16, as well as from QMiCS (820505) and OpenSuperQ (820363) of the EU Flagship on Quantum Technologies, and the EU FET Open Grant Quromorphic (828826). J. C. acknowledges the Ramón y Cajal program (RYC2018-025197-I) and the EUR2020-112117 project of the Spanish MICINN, as well as support from the UPV/EHU through the grant EHUrOPE. E. T. acknowledges financial support from Project PGC2018-094792-B-I00 (MCIU/AEI/FEDER,UE), CSIC Research Platform PTI-001, CAM/FEDER Project No. S2018/TCS-4342 (QUITEMAD-CM), and by Comunidad de Madrid-EPUC3M14.

References

[1] W. S. McCulloch and W. Pitts W, "A Logical Calculus of Ideas Immanent in Nervous Activity" , Bull. Math. Biol. vol. 5, pp. 115-133, 1943.
[2] C. M. Bishop, "Pattern Recognition and Machine Learning," Springer, 2006.
[3] J. Schmidhuber, "Deep Learning in Neural Networks: An Overview," Neural Networks, vol. 61, pp. 85–117, 2015.
[4] Y. Bengio, Y. LeCun, G. Hinton, "Deep Learning," Nature, vol. 521, pp. 436–444, 2015.
[5] J. J. Hopfield, "Neurons with graded response have collective computational properties like those of two-state neurons", Proc. Nat. Acad. Sci. vol. 81, pp. 3088-3092, 1984.
[6] G. Cybenko, "Approximation by superpositions of a sigmoidal function" Math. Control Signals Syst. vol. 2, pp. 303-314, 1989.
[7] D. O. Hebb, "The organization of behavior", Wiley, New York, 1949.
[8] F. Rosenblatt, "The perceptron: A probabilistic model for information storage and organization in the brain," Psychol. Rev. vol. 65, pp. 386-408, 1958.
[9] K. Hornik, M. Stinchcombe and H. White, "Multilayer feedforward networks are universal approximators," Neural Netw., vol. 2, pp. 359-366, 1989.
[10] D. Steinkrau, P. Y. Simard, and I. Buck, "Using GPUs for machine learning algorithms," Eighth International Conference on Document Analysis and Recognition (ICDAR'05), 2005.
[11] C. Walter, "Kryder's Law", Sci. Am. vol. 293, pp. 32-33, 2005.
[12] A. Kolmogorov, "On Tables of Random Numbers", Sankhya Ser. A. **25**, 369, 1963.
[13] M. Castells, "The Information Age: Economy, Society and Culture", Blackwell, Oxford, 1996.
[14] Y. LeCun, L. Bottou, Y. Bengio and P. Haffner, "Gradient-based learning applied to document recognition," Proc. IEEE, vol. 86, no. 11, pp. 2278–2324, 1998.
[15] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, and B. Kingsbury, " Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups" IEEE Signal Process. Mag. vol. 29, pp. 82-97, 2012.
[16] G. E. Hinton, S. Osindero, and Y. Teh, "A fast learning algorithm for deep belief nets". Neural Comput. vol. 18, pp. 1527-1554, 2006.

[17] E. G. Dada, J. S. Bassi, H. Chiroma, S. M.Abdulhamid, A. O. Adetunmbi, and O. E. Ajibuwa, "Machine learning for email spam filtering: review, approaches and open research problems," Heliyon vol. 5, pp. e01802, 2019.

[18] S. Devi, P. Malarvezhi, R. Dayana and K. Vadivukkarasi, "A Comprehensive Survey on Autonomous Driving Cars: A Perspective View". Wirel. Pers. Commun. vol. 114, pp. 2121-2133, 2020.

[19] D. Leibfried, R. Blatt, C. Monroe, C. and D. Wineland, "Quantum dynamics of single trapped ions," Rev. Mod. Phys. vol. 75, pp. 281, 2003.

[20] M. H. Devoret and R. J. Schoelkopf, "Superconducting Circuits for Quantum Information: An Outlook," Science, vol. 339, pp. 1169-1174, 2013.

[21] I. Bloch, "Ultracold quantum gases in optical lattices," Nat. Phys. vol. 1, pp. 23-30, 2005.

[22] J. L. O'Brien, A. Furusawa, and Vuckŏvić, J. Photonic quantum technologies. Nat. Photonics 3, 687-695 (2009).

[23] R. Hanson, L. P. Kouwenhoven, J. R. Petta, S. Tarucha, and L. M. K. Vandersypen, "Spins in few-electron quantum dots," Rev. Mod. Phys. vol. 79, pp. 1217, 2007.

[24] M. Dusek, N. Lutkenhaus, and M. Hendrych, "Quantum Cryptography," Prog. Opt. vol. 49, pp. 381-454, 2006.

[25] C. L. Degen, F. Reinhard, and P. Cappellaro, "Quantum sensing," Rev. Mod. Phys. vol. 89, pp. 035002, 2017.

[26] D. Loss and D. P. DiVincenzo, "Quantum computation with quantum dots," Phys. Rev. A vol. 57, pp. 120, 1998.

[27] I. Baumgart, J. M. Cai, A. Retzker, M. B. Plenio and Ch. Wunderlich, "Ultrasensitive Magnetometer using a Single Atom," Phys. Rev. Lett. vol. 116, pp. 240801, 2016.

[28] N. Timoney, I. Baumgart, M. Johanning, A. F. Varón, M. B. Plenio, A. Retzker and Ch. Wunderlich, "Quantum gates and memory using microwave-dressed states," Nature vol. 476, pp. 185, 2011.

[29] S. Weidt, J. Randall, S. C. Webster, K. Lake, A. E. Webb, I. Cohen, T. Navickas, B. Lekitsch, A. Retzker and W. K. Hensinger, "Trapped-Ion Quantum Logic with Global Radiation Fields," Phys. Rev. Lett. vol. 117, pp. 220501, 2016.

[30] S. Olmschenk, K. C. Younge, D. L. Moehring, D. N. Matsuke- vich, P. Maunz, and C. Monroe, "Manipulation and detection of a trapped Yb^+ hyperfine qubit, Phys. Rev. A vol. 76, pp. 052314, 2007.

[31] R. Puebla, Y. Ban, J. F. Haase, M. B. Plenio, M. Paternostro, and J. Casanova, "versatile Atomic Magnetometry Assisted by Bayesian Inference," arXiv:2003.02151, 2020.

[32] C. Weedbrook, S. Pirandola, R. García-Patrón, N. J. Cerf, T. C. Ralph, J. H. Shapiro, and S. Lloyd, "Gaussian quantum information," Rev. Mod. Phys. vol. 84, pp. 621, 2012.

[33] N. Gisin and R. Thew, "Quantum communication," Nat. Photonics vol. 1, pp. 165–171, 2007.

[34] F. Arute, K. Arya, R. Babbush, et al., "Quantum supremacy using a programmable superconducting processor," Nature vol. 574, pp. 505–510, 2019.

[35] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe and S. Lloyd, "Quantum machine learning," Nature, vol. 549, pp. 195-202, 2017.

[36] M. Schuld, I. Sinayskiy, and F. Petruccione, "An introduction to quantum machine learning," Contemp. Phys. vol. 56, pp. 172-185, 2015.

[37] E. Magesan, J. M. Gambetta, A. D. Córcoles and J. M. Chow, "Machine Learning for Discriminating Quantum Measurement Trajectories and Improving Readout," Phys. Rev. Lett. vol. 114, pp. 200501, 2015.

[38] W. Liu, J. Huang, Y. Li, H. Li, C. Fang, Y. Yu and G. Zeng, "Parameter estimation via weak measurement with machine learning," J. Phys. B: At. Mol. Opt. Phys. vol. 52, 045504, 2019.

[39] G. Torlai, G. Mazzola, J. Carrasquilla, M. Troyer, R. Melko, and G. Carleo, "Neural-network quantum state tomography," Nat. Phys. vol. 14, pp. 447, 2018.

[40] A. Rocchetto, S. Aaronson, S. Severini, G. Carvacho, D. Poderini, I. Agresti, M. Bentivegna, and F. Sciarrino, "Experimental learning of quantum states," Sci. Adv. vol. 5, pp. eaau1946, 2019.

[41] S. Yu, F. Albarran-Arriagada, J. C. Retamal, Y.-T. Wang, W. Liu, Z.-J. Ke, Y. Meng, Z.-P. Li, J.-S. Tang, E. Solano, L. Lamata, C.-F. Li, and G.-C. Guo, "Reconstruction of a photonic qubit state with reinforcement learning," Adv. Quantum Technol. vol. 2, pp. 1800074, 2019.

[42] C. Harney, S. Pirandola, A. Ferraro and M. Paternostro, "Entanglement classification via neural network quantum states," New J. Phys. vol. 22, pp. 045001, 2020.

[43] J. Gray, L. Banchi, A. Bayat and S. Bose, "Machine-learning- assisted many-body entanglement measurement," Phys. Rev. Lett. vol. 121, pp. 150503, 2018.

[44] J. Walln ofer, A. A. Melnikov, W. D ur and H. J. Briegel, "Machine learning for long-distance quantum communication," PRX Quantum vol. 1, pp. 010301, 2020.

[45] J. Wang, S. Paesani, R. Santagati, S. Knauer, A. A. Gentile, N. Wiebe, M. Petruzzella, J. L. O'Brien, J. G. Rarity, A. Laing and M. G. Thompson, " Experimental quantum Hamiltonian learning," Nat. Phys. vol. 13, pp. 551, 2017.

[46] A. Nagy and V. Savona, "Variational quantum monte carlo method with a neural-network ansatz for open quantum systems," Phys. Rev. Lett. vol. 122, pp. 250501, 2019.

[47] M. J. Hartmann and G. Carleo, "Neural-network approach to dissipative quantum many-body dynamics," Phys. Rev. Lett. vol. 122, pp. 250502, 2019.

[48] F. Vicentini, A. Biella, N. Regnault and C. Ciuti, "Variational Neural-Network Ansatz for Steady States in Open Quantum Systems," Phys. Rev. Lett. vol. 122, pp. 250503, 2019.

[49] A. Lumino, E. Polino, A.S. Rab, G. Milani, N. Spagnolo, N. Wiebe and F. Sciarrino, "Experimental Phase Estimation Enhanced by Machine Learning," Phys. Rev. Appl. vol. 10, pp. 044033, 2018.

[50] P. Palittapongarnpim and B. Sanders, "Robustness of quantum enhanced adaptive phase estimation," Phys. Rev. A vol. 100, pp. 012106, 2019.

[51] H. Xu, J. Li, L. Liu, Y. Wang, H. Yuan and X. Wang, "Generalizable control for quantum parameter estimation through reinforcement learning," npj Q. Info. vol. 5, pp. 82, 2019.

[52] Y. Peng and H. Fan, "Feedback ansatz for adaptive feedback quantum metrology training with machine learning," Phys. Rev. A vol. 101, pp. 022107, 2020.

[53] V. Cimini, I. Gianani, N. Spagnolo, F. Leccese, F. Sciarrino and M. Barbieri, "Calibration of Quantum Sensors by Neural Networks," Phys. Rev. Lett. vol. 123, pp. 230502, 2019.

[54] S. Kak, "On Quantum Neural Computing," Inform. Sci. vol. 83, pp. 143-160, 2015.

[55] K. Mitarai, M. Negoro, M. Kitagawa, and K. Fujii, "Quantum circuit learning," Phys. Rev. A vol. 98, pp. 032309, 2018.

[56] E. Farhi and H. Neven, "Classification with Quantum Neural Networks on Near Term Processors," arXiv:1802.06002, 2018.

[57] Y. Cao, G. G. Guerreschi, G. G. and A. Aspuru-Guzik, "Quantum Neuron: an elementary building block for machine learning on quantum computers," arXiv:1711.11240, 2017.

[58] A. Pérez-Salinas, A. Cervera-Lierta, E. Gil-Fuster, and J. I. Latorre, "Data re-uploading for a universal quantum classifier," Quantum vol. 4, pp. 226, 2020.

[59] K. H. Wan, O. Dahlsten, H. Kristjánsson, R. Gardner and M. S. Kim, "Quantum generalisation of feedforward neural networks," npj Quantum Inf. vol. 3, pp. 36, 2017.

[60] N. Killoran, T. R. Bromley, J. M. Arrazola, M. Schuld, N. Quesada and S. Lloyd, " Continuous-variable quantum neural networks," Phys. Rev. Research vol. 1, pp. 033063, 2019.

[61] E. Torrontegui and J. J. García-Ripoll, "Unitary quantum perceptron as efficient universal approximator," Europhys. Letts. vol. 125, pp. 30004, 2019.

[62] Y. Ban, X. Chen, E. Torrontegui, E. Solano and J. Casanova, "Speeding up quantum perceptron via shortcuts to adiabaticity", Sci. Rep. vol. 11, pp. 5783, 2021.

[63] Y. Ban, E. Torrontegui, and J. Casanova, "Quantum neural networks with multi-qubit potentials", arXiv: 2105.02756, 2021.

[64] Y. Ban, J. Echanobe, Y. Ding, R. Puebla and J. Casanova, " Neural-network-based parameter estimation for quantum detection," arXiv: 2012.07677, 2020.

[65] J. R. Maze, P. L. Stanwix, J. S. Hodges, S. Hong, J. M. Taylor, P. Cappellaro, L. Jiang, M. V. Gurudev Dutt, E. Togan, A. S. Zi- brov, A. Yacoby, R. L. Walsworth and M. D. Lukin, "Nanoscale magnetic sensing with an individual electronic spin in diamond," Nature vol. 455, pp. 644-647, 2008.

[66] J. M. Taylor, P. Cappellaro, L. Childress, L. Jiang, D. Budker, P. R. Hemmer, A. Yacoby, R. Walsworth and M. D. Lukin, "High- sensitivity diamond magnetometer with nanoscale resolution," Nat. Phys. vol. 4, pp. 810-816, 2008.

[67] X. Chen, E. Torrontegui and J. G. Muga, "Lewis-Riesenfeld invariants and transitionless quantum driving," Phys. Rev. A vol. 83, pp. 062116, 2011.

A Survey on Quantum Computer Simulators

Zakaria Abdelmoiz Dahi ‡§, Enrique Alba †, Rodrigo Gil-Merino ‡, Francisco Chicano † and Gabriel Luque †
‡ *Department Lenguajes y Ciencias de la Computacion, E.T.S.I. Informática, University of Malaga, Spain*
† *ITIS Software, Edificio Ada Byron, University of Malaga, Spain,*
§ *Department IFA, Faculty NTIC, Constantine 2 University, Algeria*
zakaria.dahi@{uma.es, univ-constantine2.dz}, {eat, chicano, gabriel}@lcc.uma.es, gilmerino@uma.es

***Abstract*—Quantum computers are unique systems based on peculiar properties from quantum physics, such as entanglement and superposition that allow them to provide unique computational performances. Quantum computing is meant to be revolutionary in many senses and fields. Some quantum machines have already been devised, but their accessibility or affordability to large commercialisation is yet to come. Meanwhile, a consistent plethora of quantum computer simulators are provided for users to run quantum programs on classical machines. These same programs could be later also executed in real quantum computers. This promising alternative allows practitioners to get beyond this impasse and evolve research in quantum techniques' design. Nonetheless, simulators differ from one another in many overwhelming aspects that harden classifying and profiling them and also choosing the most adequate one given a specific application purpose, programming or quantum computing paradigms. Considering these facts, our work presents a literature review of the existing quantum computer simulators. As far as we know, we are the first to perform such literature review: analyse and include 149 simulators in it and consider up to 10 comparison metrics including 21 programming languages/frameworks and also web, desktop and hybrid simulators. Our work offers several contributions by: `1)` providing a clear and encompassing repository that will allow users making appropriate choices of simulators, `2)` providing the research community with an up-to-date listing of advances in quantum computer simulation and `3)` opening new perspectives on how to build better future quantum computer simulators.**

***Index Terms*—Quantum Computer Simulators, Quantum Computing, Quantum Computers.**

I. Introduction

Quantum computers are computationally-empowered systems that take advantage of unique features such as entanglement and superposition provided by quantum mechanics [5]. They are expected to make a shift in software engineering and also deliver computational advances with endless applications [1]. Their use is awaited to be ground-breaking in so many ways. Not long ago real-world quantum machines did not exist yet and still up till today, the existing ones are either not affordable or not accessible at all for regular users (e.g. Sycamore QPU [1]). Considering that there is a paramount need for devising techniques that unleash quantum machines' power, as an alternative, while waiting for large and commercial quantum computers, classical computers capacities are being leveraged to simulate quantum computers with the help of *Quantum Computer Simulators* (QCS) [3]. Nonetheless, one should not confuse QCSs with quantum simulators like those described by Richard Feynman [2], which are real quantum systems. Also, one should bear in mind that QCSs can run quantum programs, like real quantum devices do, but they might run into limits such as memory and computation time [1].

Nowadays, a substantial variety of QCSs exist, where each one differs in several aspects (e.g. open accessibility, quantum system and language paradigms, number of qubits, etc.). Such massive configurations' variety poses several problems for picking the most adequate QCS considering a given constraint (e.g. application purpose) and also allowing the research community to keep track of the advances made in this field so as to evolve towards better future QCSs. To the best of the authors' knowledge, no work has reviewed QCSs with enough depth to afford the previous challenges. Nonetheless, one could mention the work done in [3] that dedicates a section to QCSs, although that work did not target QCSs. In addition, another interesting listing of QCSs can be found on the internet [1,2], but they are simple enumerations of QCSs not providing any in-depth technical details or classification methodology. Moreover, it provides a deprecated listing of QCSs that does not exist any more and whom the affiliated projects have been shutdown (e.g. `Fraunhofer QCS`, `GQC`, `Quantum Walks`, etc.) or even for those still active (e.g. `Davy Wybiral`) or inactive (e.g. `VirtualQC`), the provided links are incorrect or not working. Moreover, some QCSs are even programming languages (e.g. `Q-gol`, `LanQ`, `QCL`, `QWIRE`, `QASM`, `Quipper`, etc.), so one wonder if they should be classified as QCSs. Finally, some QCSs are said to be GUI-based, while actually, much more are. Also, it might happen that the same QCS is cited with new and old versions (e.g. `Quantum Fog`).

In our work we provide a comprehensive taxonomy by considering a substantial set of QCSs, various sets of comparison metrics that are important and relevant to QCSs' engineering. Technically, we conduct a systematic and encompassing literature review of the QCSs advances [4]. This is done by classifying and analysing the existing QCSs. We consider the aforementioned listings as a partial building-bricks of our work. Our contributions stands in being the first to `1)` dedicate a complete work to profiling QCSs, including 149 QCSs and `2)` consider up to 10 comparison metrics, 21 programming languages/frameworks and web, desktop and hybrid simulators.

[1]QCSs List (1): https://quantiki.org/wiki/list-qc-simulators
[2]QCSs List (2): https://qosf.org/project_list/

The remainder of the paper is as follows. In Section II, we provide some definitions and nomenclature to be used throughout our survey. Section III introduces the taxonomy and analysis of the reviewed QCSs. Sections IV and V present some final thoughts and suggestions to build better future QCSs and also list interesting application domains to be explored. Finally, Section VI concludes our paper.

II. Nomenclature and Selection Criteria

The quantum computing field contains a number of terms that are referring to different concepts, but that could, in some cases, be confused with each other. We can cite particularly quantum simulators, quantum computer simulators, and quantum programming languages. Making such distinction turns to be quite important particularly when looking at the quantum technology and software stacks given in [6]. Thus, in order to help introduce a clear and coherent nomenclature to be used in the literature, and also ease the understanding of our work, we will define here and we encourage future definitions of the three previously-cited terms to avoid any words' misuse.

We also present, in this section, the main criteria we applied to decide if a QCS should (or not) be included in our review.

A. Definitions

In this section, we present the definitions of quantum simulators, programming languages and computer simulators.

Definition 1: Quantum Simulators (QSs) are sometimes referred to as a quantum computing paradigm but technically they are task-dedicated quantum computing devices by themselves for studying a given aspect such as the model of quantum many-body mechanics. As an example of QSs, we could cite those pointed in Feynman's work [3].

Definition 2: Quantum Programming Languages (QPLs) are, in general, the set of languages that are based on classical programming paradigms (e.g. procedural, functional, multi-paradigm, etc.) or new ones (e.g. quantum-object, circuit-based, etc.) for quantum-related applications [7].

Definition 3: quantum computer simulators are, unlike quantum simulators, software that leverages classical computers to simulate quantum computers. The QCSs can be seen as a set of software layers that empowers the simulation of real quantum devices such as quantum simulators and this via quantum programming languages [3].

B. QS vs QPL vs QCS

Regarding the above-cited explanations, our work is about quantum computer simulators and not on QSs nor QPLs. Although the three might overlap in some cases, they should not be confused nor interchangeably used even if some works do it by making debatable statements. Indeed, works such as [7] states that quantum simulators cannot replace quantum programming languages. Moreover, the authors classify `QuantumOptics.jl` as a multi-paradigm quantum programming language, but they refer to it as an open quantum system, which is confusing. In addition, most QPLs reviewed in their work are based on classical ones (e.g. Object-oriented syntax, C and C++ compilers, etc.) and therefore it is not clear how a QPL has (or not) been classified as quantum-material. For example, the authors stated that `Quantum Language Q` is "*not a quantum programming language, but its library is written in C++*", although they do classify it as a QPL. Finally, they refer to `LIQUi|>` as a QPL, but it is actually a tool-suite for quantum computing that eventually could include a QPL.

C. QCSs' Sources, Inclusion and Exclusion Criteria

To the best of our knowledge, no guideline exists on what are the components'/software layers (e.g. compiler, circuit mapper/optimizer, etc.) of a typical QCS's. Doing so go beyond the scope of our paper. Also, even if most QCSs are open source, no details are given about their constituents, how they work, etc. So, as a first effort to review the existing QCSs, we have set some preliminary general criteria to decide which QCS should (or not) be included in our survey. Concretely, we integrated all QCSs that:

- Have been used in research-related works (e.g. theses, journal/conference papers, technical reports, etc.) or commonly cited in quantum-related resources (e.g. specialised magazines, fora, etc.).
- Implement basic or advanced qubits' manipulations.
- Implement basic or advanced gates' applications.
- Can be used online, via a desktop installer, or both.

More efforts are needed to establish standards of QCSs to have a clear definition of their essential components, workflow, etc. This work is a first step towards a more fine-selected QCSs and QCSs' implementation norms. Also, we would like to mention that most QCSs have not been included in officially published works, so most of our resources and references will be link-based (see Appendix A). The sources from where the studied QCSs have been extracted are:

- Academic publishers: e.g. Springer, IEEE, Elsevier, etc.
- Quantum corporations: e.g. IBMQ, D-wave, etc.
- Code's hosting platforms/pages: e.g. GitHub, personal web pages, etc.

Using the above cited-criteria, we have analysed 149 QCSs that we later filtered to 140 ones by neglecting those that we judged are not actually QCSs. After this, we further filtered the 140-QCSs-list to 100 QCSs by discarding all those who are not currently accessible (links and projects non-existent).

III. Comparison Criteria and Taxonomy of QCSs

In this section, we provide a review, analysis and taxonomy of 100 quantum computer simulators according to 10 comparison metrics. Three main families of QCSs are analysed in our work: web-based, desktop-based and hybrid (web-and-desktop-based) QCSs, where the first are online web services delivering a QCS and the second are QCSs that require offline pre-installation and execution without internet connection. For all the QCSs' classes, we performed the taxonomy according to 8 metrics: `Full-stack`, `#Qubit(s)`, `#Gate(s)`, `#Shot(s)`, `Application(s)`, `Project Status`, `Open Access` and `Open Source`.

These metrics are key features that are common, important, application-dependent and rule most QCSs' strengths and applicability. In addition, for the desktop and hybrid QCSs, we considered 2 additional comparison metrics: `programming languages` and `GUI-based` (see Figure 1). We use 21 programming languages/frameworks during the classification since it is the number we found after having extracted all the desktop and hybrid QCSs we found. We did not include the two supplementary metrics (language and GUI) for web-based QCSs since several programming languages/frameworks and APIs could be used simultaneously. One could use tools such as `wappalyzer`[3] to extract the languages involved in building a given web-based QCS. In addition, the metric `GUI-based` is not applicable to web-service-based QCSs.

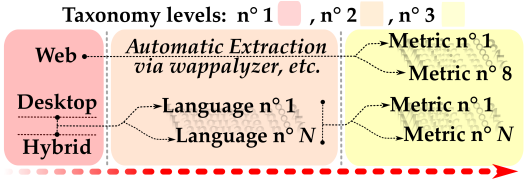


Fig. 1. Taxonomy skeleton

The 10 comparison metrics we use, have been chosen to reflect the QCSs simulation power, their accessibility and their history of activity. The metric `Full-stack` indicates if the QCS has been said (or not) to be a self-contained QCS. `#Qubit(s)` represents the number of qubits that the QCS provides. `#Gate(s)` represents the number of gates that can be used. `#Shot(s)` gives the number of times the circuit can be consecutively executed without being confronted to a given limitation (e.g. token validity, free trial ends, etc.). `Application(s)` tells us on whether the QCS is general-purpose or it has been tailored for a given field of application. `Project Status` reflects if contributions, updates and efforts are still going in the project that supports the QCS. `Open Access` indicates if the QCS is freely available (no payment required) or accessible upon payment. `GUI-based` indicates if the QCS includes a graphical-user interface or not, and `Open Source` shows if the QCS's code is freely available. `Language` indicates what programming language is used to implement the QCS. When reviewing the QCSs, we faced three main obstacles: the links of some QCSs were not working (non-existent), some QCSs could not be installed due to bugs and incompatibilities, other QCSs have a code that is too fuzzy and long to review. If the information could not be obtained or confirmed, we indicate "Unknown". Also, we use "–" for the attributes of a QCS that is found non-existent for a given reason. Indeed, most of the QCSs do not provide tutorials, documentation or insight of their use or implementation.

Figure 2 shows some statistics on the distribution of QCSs according to the platform: web, desktop and hybrid, including the programming languages they are based on. The bars' colour represents a class or language. The 1st and 2nd bars of the same colour indicate how many QCSs are considered before and after filtering, respectively. The latter is performed considering the inclusion/exclusion criteria in Section II-C. One can note that desktop-based are the most widely-spread ones, followed by web-based and then hybrid QCSs. Regarding desktop QCSs, those based on C, C# and C++ are the most popular ones.

The QCSs' list in Table II is organised per programming language (those with more QCSs to those with the least), while Tables I and III do not follow any ordering criterion. In Tables I-III, grey-shaded cells represent the best QCS(s) according to a given metric.

A. Web-based Quantum Computer Simulators

Table I represents the taxonomy of web-based QCSs, where 37.5% and 87.5% QCSs are open access and open source, respectively. One can note that most QCSs are application-tailored and also allow using a relatively high number of qubits and gates, although no much information is provided on their implementation. We also found that `Quirk` is among the top web-based QCSs. It provides a clear graphical QCS, it is easy to use via drag and drop functionality, it puts no limits on the number of qubits or shots to be used, it provides a large set of quantum gates, it provides both a video and written tutorials on how to use the QCS and it is open source.

B. Desktop-based Quantum Computer Simulators

Table II regroups the taxonomy of desktop-based QCSs, where 98.86%, 97.72% and 17.04% are open access, open source and have GUI, respectively. Regarding C/C++ QCSs, we found that `QuEST` is among the best QCSs because it provides many qubits and gates to use, it has a substantial written and video documentation and it does an efficient leveraging of the machines' CPU/RAM/network/GPU capacities. As to Java-desktop-based QCSs, we find that `jQuantum` is a promising simulator to use. Moving to Python-based QCSs, one can state that `Pyquil` is an interesting QCS to use considering the large plethora of applications and documentations it provides. For the remaining programming languages, it is hard to make firm conclusions on the usefulness of one QCS rather than others considering that not much information and insights (e.g. number of qubits, gates, shots, etc.) are given about most QCSs.

C. Hybrid Quantum Computer Simulators

Table III presents a taxonomy of hybrids QCSs, where 100%, 75% and 25% are open access, open source and have GUI, respectively. All these QCSs provide substantial documentation, quantum gates and qubits to be used that suit both industrial and research purposes. Nonetheless, we found that most hybrid QCSs are oriented towards fee-based QCS-services, which restricts their use at some point. Also, although we enumerate only QCSs, one should stress that other actors of the QC community such as `D-wave` and `IonQ` are more oriented towards fully-quantum devices rather

[3]Wappalyzer: https://www.wappalyzer.com/

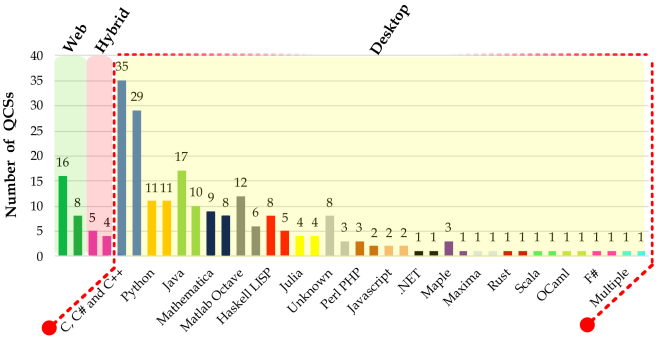


Fig. 2. Some statistics about quantum computer simulators

TABLE I
TAXONOMY OF WEB-BASED QUANTUM COMPUTER SIMULATORS

QCS Name	Full-stack	#Qubit(s)	#Gate(s)	#Shot(s)	Purpose(s)	Project Status	Open Access	Open Source
Factor 15 Circuit	Unknown	4	2	1	Shor's Algorithm	Unknown	Yes	No
Quantum Computing Playground	Unknown	22	21	1	Shor's and Grover's Algorithms, etc.	Unknown	Yes	Yes
Quantum Programming studio	Unknown	Unlimited	34	>= 1	Quantum circuit simulation	Active	Yes	No
Davy Wybiral QCS	Unknown	10	14	Unknown	Qubits manipulations	Unknown	Yes	Yes
Qubit Workbench	Unknown	Free: 4, Non-free: > 100	17	Free: > 9000, Non-free: Unknown	Qubits manipulations	Active	Yes/No	No
Quirk	Unknown	Unlimited	44	Unknown	Multiple	Unknown	Yes	Yes
Quantum Search Applet	Unknown	Unknown	Unknown	Unknown	Shor's algorithm	Unknown	Yes	No
BackupBrain	Unknown	Unlimited	9	>= 1000	Quantum circuit simulation	Active	No	No

than QCSs. Thus, they provide access to hybrid or quantum devices/algorithms and this via both web and desktop toolkits.

IV. Desired Features in Future QCS Design

Considering Section III, it can be seen that each QCS (or its category) has strengths and weaknesses. Thus, better QCSs could be built by summing up the strengths and discarding the weaknesses of all of them. We present here a set of desired features that could achieve this by: `1)` online vs desktop accessibility, `2)` interactions and manipulations, `3)` programming language paradigms, `4)` quantum computing paradigms, `5)` full-stack QCSs and `6)` software engineering principles. Our suggestions can be expanded to further aspects using more in-depth details. However, this goes beyond the scope of this paper. Also, one should keep in mind that future promising QCSs are yet to come such as `cuQuantum SDK` of nvidia, which leverage GPU performances of classical machines and also opens new perspectives in this axis.

A. Hybrid Accessibility and Execution Mode

Most QCSs provide one-handed and very constrained accessibility through webpages, while others require standalone installations. In this case, hybrid QCSs (e.g. IBMQ Experience) appear as a promising alternative that allows users with no internet access to run their programs at any moment using their personal machines, while users that have no access to desktop QCSs could use online QCSs. Moreover, synchronisation between both sides could allow users to switch back-and-forth between online and desktop QCS.

B. Simplified Interaction and Manipulation

Quantum computing users have different backgrounds. Thus, a QCS should provide manipulation tools that go with the users' profiles and expertise levels. For instance, providing graphical online and desktop QCS's features such as drag-and-drop (e.g. IBMQ, QuWire, etc.) could help academics, students and non-experts get familiar with quantum computing for teaching purposes and this without the need of programming skills. Also, automated circuit-based algorithm or automatic-problem formulation could be useful as well to avoid novice users getting too fast into technicality that might jeopardise their use of the QCS.

C. Programming Language and Quantum System Paradigms

Quantum devices are based on paradigms that rule the way they are used/executed. This ranges from problem formulation, problem mapping, algorithm declaration, algorithm mapping, quantum compilation and the used quantum simulator itself. These aspects are more related to advanced users that wish to control sophisticated aspects of the simulation. Therefore, providing unified or a multi-paradigm QCS (e.g. `QRBG`) could have several advantages such as the possibility of executing, with a reasonable change, the quantum program on several quantum computers supporting each a different paradigm. Also, each QCS is based on a particular programming language and paradigm. Thus, the variety of languages and paradigms handled by a given QCS is also a key factor to consider.

D. Full-stack Quantum Computer Simulator

Some QCSs emphasize on certain aspects and purposes of quantum computing rather than others (e.g. optimisation problem-solving, error correction, circuit optimisation, etc.). Such specialisation greatly affects the way the QCS is designed and also the range of its use. It is clear that the application domains and purposes are too large to be all integrated into a single QCS, but gathering the main functionalities could help to unify the research efforts and further comparisons between

TABLE II
TAXONOMY OF DESKTOP-BASED QUANTUM COMPUTER SIMULATORS

Language	QCS Name	Full-stack	#Qubit(s)	#Gate(s)	#Shot(s)	Purpose(s)	Project Status	Open Access	Open Source	GUI
C, C# and C++	**Intel Quantum Simulator (IQS, former qHiPSTER)**	Unknown	> 2	> 4	Unknown	Qubit's manipulations, problem solving, etc.	Active	Yes	Yes	No
	staq	Yes	Unknown	Unknown	>=1	Synthesis, transformation, optimization and compilation of	Active	Yes	Yes	No
	QuEST	Yes	45	29	>=1	Various	Active	Yes	Yes	No
	Scaffold/ScaffCC	No	–	–	–	–	–	Unknown	Yes	No
	Qrack	Unknown	> 32	Unknown	>=1	Quantum bit and gate simulator	Active	Yes	Unknown	No
	QX Simulator	Unknown	> 17	18	>=1	Quantum circuit simulation	Uknwon	Yes	Unknown	Yes
	Quantum++	Unknown	25	Unknown	>=1	General–purpose quantum computing library	Active	Yes	Yes	No
	QMDD	Unknown	Unknown	Unknown	>=1	Efficient representation and manipulation of quantum functionality	Unknown	Yes	Unknown	No
	CHP	Unknown	Unknown	4	>=1	simulator of stabilizer circuits	Inactive	Yes	Unknown	No
	libquantum (C)	–	Unknown	3	>=1	Quantum mechanics and quantum computing	Inactive	Yes	Yes	Yes
	Q++	Unknown	Unknown	Unknown	>=1	Simulating quantum computation	Inactive	Yes	Yes	Unknown
	QCLib	Unknown	20	Unknown	>=1	Classical simulation of realistic quantum computations	Inactive	Yes	Yes	Unknown
	QDD	Unknown	Unknown	Unknown	>=1	Shor's Algorithm	Unknown	Yes	Yes	No
	QGAME	Unknown	Unknown	Unknown	>=1	Quantum Algorithms	Unknown	Yes	Yes	Yes
	qsims	Unknown	Unknown	>= 2	>=1	Quantum computing in addressable optical lattices	Inactive	Yes	Yes	No
	QTM simulator	Unknown	Unknown	Unknown	>=1	Quantum Turing Machine Simulator	Inactive	Yes	Yes	–
	Quantum Computer Simulator	Unknown	Unknown	Unknown	>=1	Quantum computer simulator	Unknown	Yes	Yes	–
	Quantum Construct (qC++) / New quantum toolkit	Unknown	Unknown	Unknown	>=1	Quantum mechanical toolkit and 3D viewer	Unknown	Yes	Yes	Yes
	Quantum Network Computing	Unknown	Unknown	Unknown	>=1	Quantum computer simulations	Unknown	Yes	Yes	Unknown
	Qubiter	Unknown	Unknown	Unknown	>=1	Quantum computer simulations	Inactive	Yes	Yes	No
	QuCoSi	Unknown	Unknown	Unknown	>=1	Quantum computer simulations	Unknown	Yes	Yes	Unknown
	QuIDDPro	Unknown	40	Unknown	>=1	Generic quantum simulation	Inactive	Yes	No	–
	QWalk	Unknown	Unknown	Unknown	>=1	Simulator of quantum walks for one- and two-dimensional lattices	Active	Yes	Yes	Unknown
	Shor's Algorithm Simulation	Unknown	Unknown	Unknown	>=1	Shor's and Grover's algorithms	Active	Yes	Yes	Unknown
	sqct-Single qubit circuit toolkit	Unknown	Unknown	Unknown	>=1	Exact and approximate synthesis of single qubit circuits	Active	Yes	Yes	No
	JKQ–DDSIM	Unknown	> 4	Unknown	> 1000	Quantum simulations	Active	Yes	Yes	No
	QuIDE	Unknown	Unlimited	15		Quantum computer simulations.	Inactive	Yes	Yes	Yes
	QSim / Qsimh	Unknown	>= 1	22	>= 1	Various	Active	Yes	Yes	No
	SimQubit	Unknown	32	12	>= 1	Various: algorithms, etc.	Inactive	Yes	Yes	Yes
Python	**PyQuil /Forest**	Yes	>=6	23	>=1	Various	Active	Yes	No	No
	ProjectQ	Yes	>=22	>=9	>=1	Various	Active	Yes	Yes	No
	PyQu	Unknown	Unknown	Unknown	Unknown	Unknown	Unknown	Yes	Yes	No
	QCircuits	Unknown	>=3	>=4	>=1	Various	Active	Yes	Yes	No
	qitensor	Unknown	>=1	Unknown	>=1	Study of quantum information and quantum computing	Inactive	Yes	Yes	No
	QuaEC	Unknown	>=1	>=7	>=1	Quantum error correction and fault-tolerance	Inactive	Yes	Yes	No
	Quantum Fog	Unknown	>=1	Unknown	Unknown	Quantum mechanical behavior	Active	Yes	Yes	Unknown
	Qubiter	Unknown	>= 4	>=1	>=1	Quantum circuit simulation	Active	Yes	Yes	No
	QuTiP	Unknown	>=3	32	>=1	Simulation fo dynamics of open quantum systems	Inactive	Yes	Yes	Unknown
	sparse_pauli	Unknown	Unknown	Unknown	Unknown	large, sparse Pauli operators using pairs of sets	Active	Yes	Yes	Unknown
	toqito	Unknown	Unknown	Unknown	Unknown	Study quantum information: states, channels, and measurements.	Active	Yes	Yes	Unknown
Java	**Bloch Sphere Simulator**	Unknown	Unknown	Unknown	Unknown	Bloch Sphere Visualisation	Inactive	Yes	No	Yes
	jQuantum	Unknown	15	7	>=1	Quantum circuit simulations	Active	Yes	Yes	Yes
	jSQ	Unknown	Unknown	Unknown	Unknown	Quantum cryptography	Inactive	Yes	Yes	Unknown
	LibQuantumJava (LQJ)	Unknown	2048		>=1	Quantum computing simulation	Active	Yes	Yes	No
	QuanSuite	Unknown	>= 1	>=1	>=1	Various application suite	Unknown	Yes	Yes	Unknown
	qMIPS101	Unknown	<22	10	>=1	MIPS and quantum circuit simulator		Yes	Yes	Yes
	QuSAnn (and Multiplexor Expander)	Unknown	Unknown	Unknown	Unknown	Code generator for simulated annealing	Unknown	Yes	Yes	Unknown
	Squankum	Unknown	Unknown	Unknown	Unknown	Quantum circuit simulations	Unknown	Yes	Yes	Yes
	Strange	Unknown	>=2	>=3	>=1	Creates Quantum Programs	Active	Yes	Yes	Yes
	Linear Al	Unknown	>= 1	>= 2	>= 1	Quantum information processing	Inactive	Yes	Yes	Yes
Mathematica	**QDENSITY**	Unknown	>=5	>=7	>=1	Teleportation, Shor's and Grover's algorithms	Unknown	Yes	Yes	Unknown
	qmatrix	Unknown	Unknown	Unknown	Unknown	Computation in quantum information theory	Inactive	Yes	Yes	Unknown
	Quantum	Unknown	Unknown	Unknown	Unknown	Various algorithms, applications, etc.	Inactive	Yes	Yes	Yes
	QuantumUtils	Unknown	Unknown	Unknown	Unknown	Various	Active	Yes	Yes	Unknown
	Quantum Information Programs in Mathematica	Unknown	>=4	>=9	>=1	Quantum circuit simulation	Inactive	Yes	Yes	No
	Quantum Turing Machine Simulator	Unknown	Unknown	Unknown	>=1	Quantum Turing Machine	Active	Yes	Yes	No
	QuCalc	Unknown	Unknown	Unknown	>=1	Quantum circuit simulation and problem solving	Inactive	Yes	Yes	Unknown
	QI	Unknown	Unknown	Unknown	>=1	Symbolic analysis of quantum states and operations	Active	Yes	Yes	Unknown
Matlab Octave	**M-fun for QC Progs**	Unknown	Unknown	>=1	>=1	Various	Active	Yes	Yes	No
	QC simulator	Unknown	Unknown	Unknown	Unknown	Unknown	Unknown	No	No	Unknown
	QETLAB	Unknown	>=1	Unknown	>=1	Quantum entanglement theory	Active	Yes	Yes	Yes
	QLib	Unknown	>=3	>=4	>=1	Various: entanglement, etc.	Inactive	Yes	Yes	No
	Quantum Octave	Unknown	>=1	>=1	>=1	Various: Teleportation, Shor and Grover algorithms, etc.	Active	Yes	Yes	Unknown
	Qubit4matlab	Unknown	>=20	>= 3	>=1	Quantum information/quantum optics	Inactive	Yes	Yes	Unknown
Haskell LISP	**Quacee**	Unknown	>=2	>=3	>=1	Quantum circuit simulation	Active	Yes	Yes	No
	CS 20c Project	Unknown	Unknown	Unknown	>=1	Quantum Turing machine	Inactive	Yes	Yes	No
	Haskell Simulator of Quantum Computer	Unknown	up to 100	>=4	>=1	Quantum circuit simulation	Active	Yes	Yes	No
	QIO	Unknown	>=1	>=1	>=1	Quantum computation: algorithms, etc.	Active	Yes	Yes	Unknown
	qchas	Unknown	>=1	8	>=1	Quantum Algorithms	Active	Yes	Yes	No
Julia	**QSWalk.jl**	Unknown	Unknown	Unknown	>=1	High-performance analysis of quantum stochastic walks	Active	Yes	Yes	No
	QuantumOptics.jl	Unknown	Unknown	Unknown	>=1	Various	Active	Yes	Yes	No
	QuantumWalk	Unknown	Unknown	Unknown	>=1	Models of quantum continuous and discrete walks	Active	Yes	Yes	No
	Yao.jl	Unknown	Unknown	Unknown	>=1	Empower quantum information simulation	Active	Yes	Yes	No
Unknown	**QCAD**	Unknown	7	11	Unlimited	Quantum circuit design	Inactive	Yes	No	Yes
	Quantum Computer Emulator	Unknown	16	>= 2	>= 1	Various: algorithms, hardware designs of quantum computers, etc.	Active	Yes	No	Yes
	Q-Kit	Unknown	Unlimited	22	Unlimited	Qunatum circuit simulation	Active	Yes	No	Yes
Perl PHP	**Quantum::Entanglement**	Unknown	Unknown	Unknown	Unknown	Shor's algorithm	Inactive	Yes	Yes	Unknown
	Quantum::Superpositions	Unknown	Unknown	Unknown	Unknown	Unknown	Inactive	Yes	Yes	Unknown
Javascript	**quantum-circuit**	Unknown	>= 20	49	>=1	Quantum circuit simulation	Active	Yes	No	No
	jsqis	Unknown	Unknown	Unknown	Unknown	Quantum circuit simulation	Active	Yes	Yes	No
.NET	**Quantum.NET**	Unknown	Unknown	Unknown	>=1	Quantum circuit simulation		Yes	Yes	No
Maple	**OpenQUACS**	Unknown	Unknown	Unknown	>=1	General-purpose universal Quantum Computer Simulator	Unknown	Yes	Yes	No
Maxima	**Qinf**	Unknown	Unknown	Unknown	>= 1	Various	Active	Yes	Yes	No
Rust	**QCGPU**	Unknown	Unknown	Unknown	Unknown	GPU accelerated simulation	Active	Yes	Yes	No
Scala	**VQS - Visual Quantum Simulator**	Unknown	>=4	>=3	>=1	Schrödinger full state Quantum Simulator	Active	Yes	Yes	No
OCaml	**QOCS**	Unknown	(limited) >= 1	>= 3	>= 1	Quantum circuit simulator: e.g. Shor's algorithm	Active	Yes	Yes	No
F#	**LIQUiD**	Yes	up to 30	>= 4	>= 1	Various: error correction, algorithms, etc.	Active	Yes	Yes	No
Multiple	**QRBGS**	Unknown	Unknown	Unknown	Unknown	Random number generation	Unknown	Yes	No	No

TABLE III
TAXONOMY OF HYBRID QUANTUM COMPUTER SIMULATORS

Language	QCS Name	Full-stack	#Qubit(s)	#Gate(s)	#Shot(s)	Purpose(s)	Project Status	Open Access	Open Source	GUI
Python	**Qiskit**	Yes	Up to 5000 (specific)	26	8192	Various	Active	Yes	Yes	Online (Yes), Desktop (No)
Python	**SV1**	Unknown	Up to 34	Unknown	Unlimited upon fees	Various	Active	Yes (Limited)	No	No
Python/Q#	**QDK**	Yes	Up to 30	>= 3	>= 1	Various	Active	Yes (Limited)	Yes	No
Python	**Cirq**	Unknown	20 internal and 30 external	>= 27	>= 1	Various	Active	Yes	Yes	No

the works conducted in the field of quantum computing. In this sense, a likely-to-be-generalised effort has already been done in `Terra`, `Aqua`, `Ignis` and `Aer` packages.

E. Respect of the Software Engineering Principles

To the best of our knowledge, no quantum software engineering principles have been devised yet, but still the QCSs are based on classical software. Thus, it is paramount to apply proper principles of software engineering. Indeed, authors in [3], have stated that most open access quantum computing software have not been devised in respect of software engineering bases. Thus, efforts are encouraged so that future QCSs will be built by considering such an important aspect.

V. Future QCSs' Application Domains

Even if QCSs' power is constrained by the machine they are used on, they still can be applied to several domains. We can cite as a first example, *artificial intelligence and problems' solving*. This stands in devising new hybrid or quantum algorithms that take advantage of the states' superposition and qubits' entanglement to solve intractable optimisation problems in various domains such as machine learning in artificial intelligence.

As a second QCSs' application, one could mention *quantum software engineering*. This includes software testing, hybrid quantum-classical software design, software quality assessment and classical-to-quantum software migration [6]. Finally, an interesting QCSs' application domain is *quantum machines' design*. This stands in evolving the design of quantum systems so they can reach new milestones in quantum computation. Many aspects such as quantum error correction, circuit optimisation and mapping, etc. are related to this axis.

VI. Conclusion

In this paper, we have conducted a systematic and comprehensive review of QCSs by `I)` considering 149 QCSs `II)` performing a comparison over 10 metrics, `III)` including 21 programming languages/frameworks and `IV)` web, desktop and hybrid simulators. Our work can be used to `(1)` make fast, easy and adequate QCSs' selection considering a given specific application, `(2)` allow academics and research community to keep an updated track of QCSs' engineering advances and `(3)` provide propositions for future QCSs' design and applications. We found that C and Python-based QCSs are the most spread, where `Quirk`, `QuEST`, `Pyquil` and hybrid ones are among the most promising QCSs to be used nowadays.

Acknowledgements

This research is partially funded by the Universidad de Málaga, Consejería de Economía y Conocimiento de la Junta de Andaluía and FEDER: grant number UMA18-FEDERJA-003 (PRECOG); Spanish Ministry of Science, Innovation and Universities and FEDER: contract RTC-2017-6714-5 (Eco-IoT); and TAILOR ICT-48 Network (No 952215) funded by EU Horizon 2020 research and innovation programme.

References

[1] Arute, F., Arya, K., Babbush, R., and et al. Quantum supremacy using a programmable superconducting processor. *Nature 574* (October 2019), 505–510.

[2] Feynman, R. Simulating physics with computers. *International Journal of Theoretical Physics volume 21* (June 1982), 467–488.

[3] Fingerhuth, M., Babej, T., and Wittek, P. Open source software in quantum computing. *PLOS ONE 13*, 12 (Dec 2018), e0208561.

[4] Paul, R. ACM SIGSOFT empirical standards version 0.1.0.

[5] Pednault, E., Gunnels, J. A., Nannicini, G., Horesh, L., and Wisnieff, R. Leveraging secondary storage to simulate deep 54-qubit sycamore circuits, 2019.

[6] Piattini, M., Serrano, M., Perez-Castillo, R., Petersen, G., and Hevia, J. L. Toward a quantum software engineering. *IT Professional 23*, 1 (2021), 62–66.

[7] Sunita, G., Maryam, G., and Amir, A. Quantum programming language: A systematic review of research topic and top cited languages. *Archives of Computational Methods in Engineering volume 28* (December 2021), 289–310.

Appendix

This appendix includes the links to the QCSs we have analysed in our work. The links are organised per class (web, desktop and hybrid) and further by programming languages. The QCSs order is the same as they appear in Tables I-III. The list is also maintained online to keep it updated on the long term (see final link in the table).

TABLE IV
Quantum computer simulators' links

Language	QCSs' Link according to QCS order
Web-based	
-	http://web.archive.org/web/20051214071130/http://www.isi.edu/acal/quantum/simulate.html http://www.quantumplayground.net/#/home https://davywybiral.blogspot.com/2012/12/quantum--circuit--simulator.html https://elyah.io/product https://algassert.com/quirk#circuit={\%22cols\%22:[]} https://joanv.me/qucomp/qucompApplet.html https://backupbrain.github.io/quantum-compiler-simulator/ https://quantum-circuit.com/docs,https://quantastica.com/
Desktop-based	
C, C# and C++	https://github.com/iqusoft/intel-qs https://github.com/softwareqinc/staq https://quest.qtechtheory.org/ https://github.com/epiqc/ScaffCC https://vm6502q.readthedocs.io/en/latest/index.html http://quantum-studio.net/ https://github.com/softwareQinc/qpp http://www.informatik.uni-bremen.de/agra/eng/qmdd.php https://www.scottaaronson.com/chp/ http://www.libquantum.de/ http://sourceforge.net/projects/qplusplus/ https://www.quantware.ups-tlse.fr/QWLIB/ http://thegreves.com/david/QDD/qdd.html http://faculty.hampshire.edu/lspector/qgame.html http://qsims.sourceforge.net/ http://web.archive.org/web/20050923134721/http://www.lri.fr/~durr/Attic/qtm/ http://www-imai.is.s.u-tokyo.ac.jp/~tokunaga/QCS/simulator.html https://sourceforge.net/projects/qcplusplus/ https://sourceforge.net/projects/qnc/ http://www.ar-tiste.com/qubiter.html https://sourceforge.net/projects/qucosi/ http://vlsicad.eecs.umich.edu/Quantum/qp/ http://www.cos.ufrj.br/~franklin/qwalk/ https://quantum-algorithms.herokuapp.com/ https://github.com/vadym-kl/sqct https://github.com/iic-jku/ddsim http://www.quide.eu/ https://github.com/quantumlib/qsim https://sourceforge.net/projects/simqubit/
Python	https://pyquil-docs.rigetti.com/en/stable/start.html http://projectq.ch/ https://code.google.com/archive/p/pyqu/ http://www.awebb.info/qcircuits/index.html http://stahlke.org/dan/qitensor/ http://www.cgranade.com/python-quaec/ https://github.com/artiste-qb-net/quantum-fog https://github.com/artiste-qb-net/qubiter http://qutip.org/ https://github.com/bcriger/sparse_pauli https://vprusso.github.io/toqito/
Java	https://eecs.ceas.uc.edu/~cahaymm/blochsphere/ http://jquantum.sourceforge.net/ https://sourceforge.net/projects/simu-quantique/ https://github.com/gbanegas/libQuantumJava http://www.ar-tiste.com/QuanSuite.html http://institucional.us.es/qmipsmaster/ http://www.ar-tiste.com/qusann.html http://jeffwass.github.io/Squankum/ https://github.com/redfx-quantum/strange http://linearal.sourceforge.net/#Home
Mathematica	http://www.pitt.edu/~tabakin/QDENSITY/ https://library.wolfram.com/infocenter/MathSource/1893/ http://homepage.cem.itesm.mx/lgomez/quantum/index.htm https://github.com/QuantumUtils/quantum-utils-mathematica https://quantum.phys.cmu.edu/QPM/ https://github.com/sdiemer/QTMSim https://library.wolfram.com/infocenter/MathSource/657/ https://github.com/iitis/qi
Matlab Octave	http://www.ar-tiste.com/m-fun/m-fun-index.html http://www-m3.ma.tum.de/Software/QCWebHome http://www.qetlab.com/Main_Page https://www.tau.ac.il/~quantum/qlib/qlib.html https://github.com/iitis/quantum-octave http://bird.szfki.kfki.hu/~toth/qubit4matlab.html
Haskell LISP	https://github.com/kat31416/quacee http://web.archive.org/web/20011207175140/www.cs.caltech.edu/~thoth/code.html http://web.archive.org/web/20010803034527/http://www.numeric-quest.com/haskell/QuantumComputer.html http://hackage.haskell.org/package/QIO https://hackage.haskell.org/package/qchas
Julia	https://github.com/iitis/QSWalk.jl https://qojulia.org/ https://github.com/iitis/QuantumWalk.jl https://github.com/QuantumBFS/Yao.jl
Unknown	http://qcad.osdn.jp/ http://www.compphys.org/QCE/ https://sites.google.com/view/quantum-kit/home
Perl PhP	https://metacpan.org/release/AJGOUGH/Quantum-Entanglement-0.32 https://metacpan.org/release/LEMBARK/Quantum-Superpositions-2.02
JavaScript	https://www.npmjs.com/package/quantum-circuit https://github.com/garrison/jsqis
.NET	https://github.com/phbaudin/quantum-computing
Maple	http://web.archive.org/web/20060116174553/http://userpages.umbc.edu/~cmccub1/quacs/quacs.html
Maxima	https://github.com/jlapeyre/qinf
Rust	https://qcgpu.github.io/
Scala	https://github.com/gmenier/VisualQuantumSimulator/wiki/Introduction
OCaml	https://github.com/dillanchang/QOCS
F#	https://tinyurl.com/Liquid-qcs
Multiple	http://random.irb.hr/
Hybrid	
-	https://quantum-computing.ibm.com/ https://aws.amazon.com/fr/braket/ https://azure.microsoft.com/fr-fr/resources/development-kit/quantum-computing/ https://quantumai.google/cirq
Permanent Link to QCSs' List	
https://github.com/Zakaria-Dahi/QCSs-List.git	

Quantum Computing: Present and Prospects

Rodrigo Gil-Merino*, Enrique Alba†, José Francisco Chicano†, Zakaria Abdelmoiz Dahi*‡, Gabriel Luque†
*Dept. Lenguaje y Ciencias de la Computación, E.T.S.I. Informática, Universidad de Málaga, Spain
†ITIS Software, Edificio Ada Byron, Universidad de Málaga, Spain,
‡Dept. IFA, Fac. NTIC, Constantine 2 University, Algeria
Emails:gilmerino@uma.es, eat, chicano, gabriel@lcc.uma.es, zakaria.dahi@uma.es, univ-constantine2.dz

***Abstract*—Quantum computing (QC) promises more powerful computers than classical ones and faster solutions to complex problems. Currently, there are two main paradigms in QC: quantum gate computers and quantum annealers. Both technologies are well established and face similar problems: scalability of the number of qubits, robustness of QC, and access to quantum facilities. In this work we review the basis and the state-of-the-art of these two technologies, analyse the aforementioned problems, and describe the near future challenges they face.**

***Index Terms*—Quantum Computing, Quantum Algorithms, Quantum Annealer, Quantum Error Correction, Quantum Computer Languages, Quantum Physics, Artificial Intelligence**

I. Introduction

QC is a new paradigm in computer science. It requires a fresh perspective both in programming and in hardware architecture, since several quantum physical effects must be taken into account when designing quantum software and hardware. The promise land of QC is a set of quantum algorithms (QAs) that outperform classical computation when running on quantum computers. For some classes of computational problems, this already starts becoming a reality, as with the case of the Shor's algorithm to factorize large numbers [1] or the Grover's algorithm to find an element in a unsorted list [2].

Nevertheless, although we understand the quantum nature of QC, we are still far from having multi-purpose personal quantum computers at home. There are different reasons for this: scaling a quantum computer to a large number of qubits, avoiding or significantly decrease quantum errors, increasing quantum fault tolerance, and/or developing quantum software platforms to allow programmers an easy access to all the capabilities of quantum machines.

This work is an attempt to compile the present success and challenges of QC. Our main contributions are the following:

- We describe the current difficulties to scale current quantum computers in terms of the number of qubits needed for practical applications.
- We analyse the current and near future strategies to decrease quantum errors depending on quantum architectures.
- We compare the present and prospects of the two main QC approaches: quantum gate computers and quantum annealing computers,

We organised the rest of the paper in the following sections. In Section II we review the concepts of quantum physics needed to understand QC: superposition, entanglement, parallelism, quantum tunnelling, and measurement. In Section III we expound the transition from classical computing to the two main paradigms in QC: quantum gates-based computers and quantum annealing-based computers. In Section IV we describe the state-of-the-art of quantum computers in number of qubits and its evolution. In Section V we explain the undesired interactions of quantum computers with the environment that induce quantum errors during computational processes and the current framework for quantum error correction and fault tolerant strategies. In Section VI we sketch several possibilities to currently access quantum computers. In Section VII we describe the challenges in QC future research. In Section VIII we explain the some industrial challenges in industrial applications. In the final Section IX we provide our conclusions.

II. Basic concepts for Quantum Computing

The basic blocks of classical computing are the binary units of data called *bits*. The two possible values of a bit are usually represented as 0 and 1, which translates to different voltage levels in a digital electronic circuit.

The basic units of data in QC are the *quantum bits* (*qubits*). Physically, they are usually subatomic particles or photons. Thus, the behaviour of the qubits is better described under the odd properties of quantum physics [3]. In the following, we review four important properties related to QC: superposition, entanglement, parallelism, quantum tunnelling, and measurement.

A. Superposition

The possible states of a qubit are 0, 1 and, additionally, a "mix" of the two. Mathematically, the qubit is a vector in a Hilbert space[1], and the "mix" is expressed as a linear combination of the two basis vectors 0 and 1 of the Hilbert space. Using the *Dirac notation*[2], a qubit $|\psi\rangle$ can be written as $|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle$, where $|0\rangle$ and $|1\rangle$ are the proper vectors, and α, $\beta \in \mathbb{C}$ are the coefficients of the linear combination, called *amplitudes*. The values $|\alpha|^2$ and $|\beta|^2$ represent, according to the laws of quantum physics, the probabilities of finding the system in the state $|0\rangle$ or $|1\rangle$ after a measurement of the

[1]A Hilbert space is an extension of a vector space with the inner product defined as the vector product within the field $\mathbb{C}$ of complex numbers.

[2]The Dirac's notation represents row vectors, called *bra*, as $\langle u|$ and column vectors, called *ket*, as $|v\rangle$, so that the vector product between $\vec{u}$ and $\vec{v}$ is written as $\langle u|v\rangle$.

qubit, respectively, and satisfy $|\alpha|^2 + |\beta|^2 = 1$. This means that, in general, we can only describe the states of a quantum system (here a qubit) in terms of probabilities associated to their amplitudes. Extending the idea to n qubits, we have 2^n possible states that can be combined into a new superposed state, written as:

$$|\phi\rangle = \alpha_0 |0...00\rangle + \alpha_1 |0...01\rangle + ... + \alpha_{2^n-1} |1...11\rangle \quad (1)$$

where $\alpha_i \in \mathbb{C}$ and $\sum |\alpha|^2 = 1$.

Superposition is a key property in QC. In a classical system composed by n bits, we can store values from 0 to $2^n - 1$, but the system processes one of those 2^n values, and only one, at a time. In a quantum system, on the contrary, n qubits are able to process all the 2^n states at the same time. Thus, from an information theory point of view, the qubit is able to store more information than the classical bit. Superposition gives to QC a potential speed-up compare to classical computing.

B. Entanglement

Sometimes we cannot describe the quantum state of a particle within a set independently of the others, we say there is entanglement between the particles. An illustrative case is a pair of particles forming a quantum state of spin zero. The knowledge of the spin of one of the particles "forces" the value of the spin of the other. This pair of particles is *entangled* [4]. In the case of QC, we can explain entanglement as an special case of superposition of multiple qubits.

In general, in a quantum computer not all qubits are entangled. The entanglement of the qubits depends on the physical interconnections between them, and on the physical architecture of the quantum computer. This is an important issue when considering the purpose of the computer and the QAs it will run, as we will see in Section III-B.

C. Parallelism

Along quantum computations, a multiple qubit system $|\phi\rangle$ evolves through unitary transformations[3] [5], which are equivalent to rotations in a Hilbert space. The quantum system $|\phi\rangle$ is a superposition of all the base states of that Hilbert space. Any transformation applied to the system will transform all base states at the same time. The simultaneous transformation of all the possible states is the quantum parallelism. Parallelism is the essence of QC, and the potential source for faster algorithms compared to classical computers [6].

D. Quantum Tunnelling

Quantum tunnelling is the ability of a particle of getting across a potential barrier that is forbidden by classical physics. It is a quantum mechanics effect related to the Heisenberg uncertainty principle. The uncertainty principle establishes that there are pairs of physical quantities of quantum systems that cannot be known with accuracy at the same time, like *position* and *momentum*. Mathematically, the reason for this is that the two quantities are *conjugate variables* in the Hilbert space or, in other words, they are Fourier transforms of one another in that Hilbert space.

This mathematical view helps in understanding that, if a variable associated to a particle is known, then a conjugate variable to the first one will be in a superposition of all its possible states. This means that if, for example, we know the total energy of the particle, then the conjugate variable position[4] should be expressed as a superposition of all the possible states of position, and then some of these possible states will locate the particle at the other side of a potential barrier of energy greater than the total energy of the particle. This is another way of viewing quantum tunnelling, an effect that will explain part of the QC described in Section III-B.

E. Measurement

To measure a physical system means to assign values to the variables that describe the system. In a quantum system, a measurement modifies the state of the system. The reason is because the quantum system might be in a linear combination of all possible states but, when we measure, the measurement will reveal one, and only one, of those states. Mathematically, if the quantum system is a qubit $|\psi\rangle = \alpha |0\rangle + \beta |1\rangle$, in superposition of the states $|0\rangle$ and $|1\rangle$, after a measurement/observation of the qubit, it will end in one its base states, either $|0\rangle$ or $|1\rangle$.

The lost of superposition of a quantum system, for example a qubit or a set of them, is usually referred to as the *wave function collapse*, because the function that represents the quantum system "collapses" to one of its base states. The collapse of the wave function is a natural end in QC, because after running a QA the final step will be measuring the state of the system. Nevertheless, a set of qubits might also collapse and lose superposition for reasons beyond measurement, such as interaction with the environment, resulting in quantum errors/defaults. The interaction of the quantum system with the environment inducing loss of superposition is called *decoherence*. Preventing quantum computers from quantum decoherence and quantum errors, and increasing their fault tolerance, is a very active field in QC, and one of the big challenges that will be treated in Section V.

III. From Classical to Quantum Computers

Due to the quantum properties already discussed in Section II, QC often needs a different approach when designing algorithms to solve problems. There is not always a quantum counterpart to a classical algorithm or, if it exits, it might not be the best/most efficient solution. QC is a completely new form of programming that we can divide in two basic representative paradigms: one is based on universal quantum gates, used to build quantum circuits and QAs; the other is based on quantum annealing, where we pursue the global minimum of a target function as an optimisation problem.

[3] A transformation U is unitary if the product $UU^\dagger = I$, where $U^\dagger$ is the conjugate transpose of U and I is the identity matrix.

[4] In general, the operator total energy does not conmute with the rest of the usual operators of the system: position, momentum, potential energy and kinetic energy. The total energy is a conjugate variable of all the others [3].

These two paradigms are depicted in Figure III. From a problem definition, one could adopt two different strategies to search for a solution with QC. A first option is to design a QA, convert it into a quantum circuit, and send this circuit configuration to a quantum computer or quantum computer simulator. A second option is to express the problem in the Quadratic Unconstrained Binary Optimisation (QUBO) formalism, called *Ising form*, that consists in defining a cost function in the form of the Hamiltonian of the system, and adapt the QUBO to the hardware of quantum annealing computer. In the following subsections we further explain these concepts in more detail.

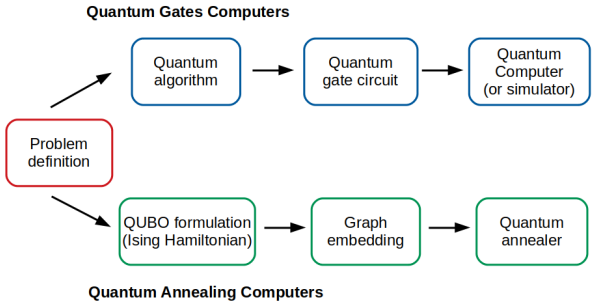


Fig. 1. The two paradigms in QC: quantum gates computers or quantum annealing computers. Although, in principle, we can formulate any problem in the two paradigms, some problems fall more naturally in one of the two.

A. Quantum Gates Computers

The transformations of the quantum states of the qubits are performed through quantum gates. This is similar to the way in which classical bits are transformed using digital gates. Since quantum states can be seen as vectors in a Hilbert space, quantum gate transformations are easily visualised as rotations of these vectors. All quantum gates are unitary transformations and they keep entanglement and superposition. A graphical view makes use of the Bloch sphere from Figure 2: a quantum state is represented by $|\psi\rangle$, written as a superposition of the two basic states in the 2-dimensional Hilbert space: $|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle$ or, alternatively, $|\psi\rangle = cos(\theta)\,|0\rangle + e^{i\phi}\sin(\theta)\,|1\rangle$, where $-\pi/2 \leq \theta < \pi/2$ and $0 \leq \phi < 2\pi$. The quantum gates can also be represented as matrices of the rotations. As an example, three basic quantum gates are shown in Table I. The *Pauli* and *Hadamard* gates act on single qubits, whereas the *controlled-NOT* (C-NOT) acts on two qubits.

In this QC paradigm, QAs must be translated into quantum circuits: a collection of quantum gates that performs the transformations on the necessary qubits. A very simple example of this is shown in Figure 3. There, two qubits are swapped using three consecutive C-NOT gates. This simple circuit is important in the implementation of quantum error correction, and important subfield in QC that we will detail in Section V.

B. Quantum Annealing Computers

The term *annealing* comes from the forges, where metalworkers heat and hammer the steel to improve its quality. This technique allows to slowly mesh the atoms of iron and other metals, resulting in layers of an stronger product. In Physics, it extends to processes that use a heat treatment to change the physical properties of a system, and later slowly cools down and evolves to an state of low inner/total energy.

A classical algorithm inspired in this concept is called *simulated annealing* (SA) [7] and has been applied to many optimisation problems. SA is a metaheuristic built in such a way that the variables of a target function are randomly modified until the function arrives at its global minimum. In general, the algorithm starts with a high energy and high temperature state of the system, and randomly moves the system to another state. If the resulting total energy is lower, then this new state is accepted and the algorithm takes the next step cooling down the system, that is, with an slightly lower temperature. The algorithm tries to find the minimum total energy of the system with the lowest temperature. The risk is the local minima, which implies larger computing time to better explore the parameter space.

Quantum annealing (QA) [8] is based on the classical simulated annealing. It is a quantum metaheuristic algorithm

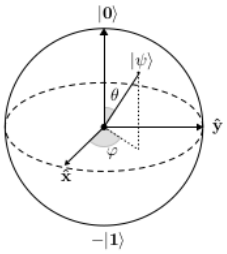


Fig. 2. The Bloch sphere. Here, the qubit $|\psi\rangle$ is not in any of the pure states $|0\rangle$ or $|1\rangle$, but in a superposition of the two.

TABLE I
THREE BASIC QUANTUM GATES, THEIR DIAGRAMS AND THEIR TRANSFORMATION MATRICES.

Name	*Circuit symbol*	*Matrix*
X - Pauli	X	$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$
Hadamard	H	$\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$
Controlled-NOT		$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$

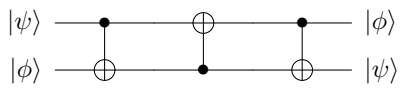


Fig. 3. An example on the use of quantum gates to swap two qubits.

for optimisation that particularly exploits the quantum property of tunnelling. In particular, the difficulties of local minima in classical simulated annealing are overcome by the quantum system tunnelling through the potential barriers without the need of returning to higher energy states. QA should find, theoretically, global minima in combinatorial optimisation problems faster than the classical simulated annealing for similar problems.

There are two main steps in QA: mapping and embedding. Mapping consists in writing the physical problem in its QUBO form, selecting the proper binary variables and constraints for the cost function. Embedding is a hardware-dependent step that links binary variables to pairs of connected qubits. Let us see these two steps further.

The QUBO formulation is a cost function, represented by an Ising Hamiltonian, with the form (e.g., [9] and references therein):

$$H_0 = -\sum_{i<j} J_{ij}\sigma_i\sigma_j - \sum_j h_j\sigma_j \qquad (2)$$

where σ_i is the binary value of the qubit i, J_{ij} is the coupling between the qubits, and h_j is the bias of each qubit j. Since not all qubits are coupled, the embedding step guarantees the connection between qubits (and so, the J_{ij} values). QA evolves adiabatically this problem Hamiltonian H_0 until it reaches its ground state, that will be the global minimum/solution.

The challenge of QA is to properly express the problem in its best QUBO form and to develop tools to help programmers to make this transformation an easy task [12].

IV. Scalability and Quantum Computing

Most of the current efforts in QC pursuit to enlarge the size of quantum computers or, from a simplistic but illustrative sight, the number of qubits available. The larger the number of qubits within a quantum computer, the higher the number of variables that a QA will handle.

The number of qubits of a quantum computer depends on the technology and architecture applied. Two flagships in the race of this scalability are the IBM [10] and D-Wave companies [11]: the first promises more than 1000-qubits at the end of 2023; the second more than 5000-qubits this year. Figure IV shows the timeline of QC in terms of the number of qubits for these two companies, which well represents the state-of-the-art of quantum computers.

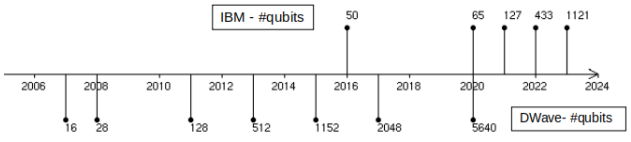


Fig. 4. Timeline of the number of qubits available in the two QC paradigms.

The increase of the number of qubits in quantum computers implies less stable machines and higher probability of errors in QC. How to control quantum errors is explained in the next Section.

V. Robustness and Quantum Computing

We refer to *robust* QC to quantum computations isolated from the environment and tolerant to hardware defects. Robust QC must guarantee that an arbitrary large quantum computation will end with success independently of the number of qubits involved and the computing time needed. There are many limitations to assure robust QC: difficulty in isolating quantum systems, impossibility of coping unknown quantum states (the no-cloning theorem), and managing the interactions between qubits at hardware level are the most prominent examples. We can divide the challenge of reaching a robust QC in two main aspects of it: quantum error correction and quantum fault tolerance. We discuss both in the rest of this Section.

A. Quantum Error Correction

Quantum error correction (QEC) is probably one of the most active fields in QC. The first QEC code was independently published by Shor [13] and Steane [14]. In general, the idea behind QCE is the use of extra qubits to "control" the ones performing the computations. This means that from the total amount of built qubits in a quantum computer, some of them must be dedicated to QEC, reducing the number of available qubits for computation. The simplest case is the transmission of a qubit that is helped with two additional ones to assure error-free transmission, that we now review mirroring [15]. We adhere to the standard convention where *Alice* transmits and *Bob* receives the transmission.

Alice wants to send a qubit to *Bob* via a noisy channel. We assume an artificial noise on the channel that randomly produces one of the two following effects on the qubit: either the state of the qubit remains unchanged, with probability $(1-p)$, or the state of the qubit changes according to the X-Pauli operator (see Table I), with probability $p < 1/2$. The state of the qubit that Alice will send can be written as $|\psi\rangle = \alpha|0\rangle+\beta|1\rangle$, and she prepares two additional qubits in the state $|0\rangle$. The state of the three initial qubits will be $|\phi\rangle = \alpha|000\rangle+\beta|110\rangle$. *Alice* applies then a C-NOT gate (see Table I) from the first qubit to the third one, producing a new state $|\psi\rangle = \alpha|000\rangle + \beta|111\rangle$. *Alice* sends the three qubits. *Bob* receives the three qubits, but he knows the channel is noisy. There are eight possibilities of the states of the qubits, from $(\alpha|000\rangle + \beta|111\rangle)$, with probability $(1-p)^3$, to $(\alpha|111\rangle + \beta|000\rangle)$, with probability p^3. *Bob* introduces two further qubits in the state $|00\rangle$, called *ancilla* qubits, to check on the effect of the noise. Then, he applies two C-NOT gates: one from the first and second qubits to the first ancilla qubit; another from the first and third qubits to the second ancilla qubit. Now *Bob* has five qubits with again eight possibilities for the states, from $(\alpha|000\rangle + \beta|111\rangle|00\rangle)$ with probability $(1-p)^3$, to $(\alpha|111\rangle + \beta|000\rangle|00\rangle)$, with probability p^3. We summarise all those states, both from Alice and Bob, in Table II to better visualise them.

Bob measures now the ancilla qubits. Possible output values are $00, 01, 10, 11$. The information obtained by this measurement is called *error syndrome*, because it is a diagnosis of

TABLE II
QUANTUM STATES AND PROBABILITIES ON TRANSMISSION.

Alice	Bob + ancilla	probability
$(\alpha\|000\rangle + \beta\|111\rangle)$	$(\alpha\|000\rangle + \beta\|111\rangle)\|00\rangle)$	$(1-p)^3$
$(\alpha\|100\rangle + \beta\|011\rangle)$	$(\alpha\|100\rangle + \beta\|011\rangle)\|11\rangle)$	$p(1-p)^2$
$(\alpha\|010\rangle + \beta\|101\rangle)$	$(\alpha\|010\rangle + \beta\|101\rangle)\|10\rangle)$	$p(1-p)^2$
$(\alpha\|001\rangle + \beta\|110\rangle)$	$(\alpha\|001\rangle + \beta\|110\rangle)\|01\rangle)$	$p(1-p)^2$
$(\alpha\|110\rangle + \beta\|001\rangle)$	$(\alpha\|110\rangle + \beta\|001\rangle)\|01\rangle)$	$p^2(1-p)$
$(\alpha\|101\rangle + \beta\|010\rangle)$	$(\alpha\|101\rangle + \beta\|010\rangle)\|10\rangle)$	$p^2(1-p)$
$(\alpha\|011\rangle + \beta\|100\rangle)$	$(\alpha\|011\rangle + \beta\|100\rangle)\|11\rangle)$	$p^2(1-p)$
$(\alpha\|111\rangle + \beta\|000\rangle)$	$(\alpha\|111\rangle + \beta\|000\rangle)\|00\rangle)$	$(p)^3$

the error. Depending on output, *Bob* performs the following actions: if 00, he does nothing; if 01, 10 or 11, he applies a X-Pauli gate to the third, second or first qubit, respectively. Finally, *Bob* keeps the most probable state and applies a C-NOT gate to it. He obtains the original qubit sent by *Alice* with probability greater than $(1-p)$ (remember that $p < 1/2$).

This code corrects a single-qubit bit-flip error. The importance of this very simple example of qubit transmission and quantum error correction is that the probability that *Bob* fails in getting the qubit sent by Alice is $O(p^2)$, whereas without error correction it would have been $O(p)$.

The development of QEC methods and frameworks is a difficult task and is one of the most important goals to the success of a robust QC for the incoming years. An obvious way of generalising the above QEC circuit is concatenating several of them to decrease even more the probability of failing in qubit manipulation. One of the challenges in concatenating quantum correction circuits is that we increase the number of qubits and quantum gates, so errors might appear more often. We treat this in the next Subsection.

B. Quantum Fault Tolerance

There is an important question in QC that we need to think of at this point: can we perform a quantum computation during an arbitrary amount of time without being overcome by noise and quantum decoherence?

The answer to this question comes from the work by [16]. The *quantum threshold theorem*, also called *quantum fault tolerance theorem* establishes that the application of QEC codes can eliminate any logical error rate to arbitrary low levels. This theorem implies that we can indeed build fault tolerant quantum computers. The general scheme for the quantum threshold theorem is given by [17]: the idea is to apply periodically an error-correction code on encoded states, so we prevent error accumulation, and creating hierarchical fault tolerant procedures.

Consequently, noise in QC is no longer an intractable problem. The challenge is to create proper QEC codes and design strategies to decrease the number of additional quantum circuits to increase fault tolerance, so that only a small fraction of all the qubits are intended to fault tolerance.

VI. CURRENT ACCESS TO QUANTUM COMPUTERS

We briefly describe in this section some of the companies in the market and universities that offer access to quantum computers.

- *Amazon Braket - Amazon*. A development environment to explore and design QAs, test them on a simulated quantum computer, and run them on your choice of different quantum hardware technologies.
- *D-Wave SDK - D-Wave Systems* It is a suite of open-source *Python* tools. Unlike the others, D-Wave machines are quantum annealers. The company offers a quantum cloud service.
- *Forest SDK - Rigetti Computing*. Based on the *Python* library *pyQuil*, with a Quantum Virtual Machine (QVM) and a compiler (*quilc*). Offers cloud access to their 32-qubits *Aspen-9* quantum computer.
- *Forge - QC Ware*. Access to D-Wave hardware as well as Google and IBM simulators. The platform offers a 30-day free trial, with 1-minute of quantum computing time.
- *IBM Q Experience - IBM*. Uses the *Qiskit* language and gives access to several quantum processors, ranging from 27 to 65 qubits.
- *LIQUi|⟩ - Microsoft*. It is a quantum computing simulation platform. The succesor to this platform is *Q♯*.
- *Quantum in the Cloud - The University of Bristol (UK)*. A free, open web interface to a 4-qubit optical quantum photonic chip with *simulation* and *experiment*.
- *Quantum in the Cloud - Tsinghua University (China)*. It is a 4-qubit quantum chip based on nuclear magnetic resonance.
- *Quantum Inspire - Qutech*. It is a cloud-based quantum computing with access to two hardware chips: a 5-qubit transmon (a type of superconducting charge qubit) quantum processor and a 2-qubit electron spin quantum processor.
- *Quantum Playground - Google*. A simulator with a simple interface, and a scripting language and 3D quantum state visualization.
- *Xanadu Quantum Cloud - Xanadu*. It is a cloud-based access to three fully programmable photonic quantum computers, with 8-qubits, 12-qubits and 24-qubits. Uses the programming suite called *Strawberry Fields*.

VII. FUTURE RESEARCH IN QC PARADIGMS

Most of the steps in the two QC paradigms moves towards larger quantum machines in terms of the number of qubits. Main companies point towards this goal: IBM promises 1000-qubit computers for 2023 and is already preparing a 1 million qubit machine for the near future; D-Wave will offer at the end of this year a 5000-qubit quantum annealer. The byproduct of this scalability in the number of qubits in both QC paradigms is clear: there is an active research in the physical construction of the qubits, more compact architectures, lower noise levels and more efficient QEC circuits.

However, quantum software development does not seem to scale as fast as quantum hardware. Nowadays there is a gap

between quantum programmers and quantum hardware, and clearly quantum software needs a boost in many aspects. High level quantum languages as transparent to the programmer as they can, testing and debugging tools and standard ways of physical problems transcription into quantum formulations are needed.

An obvious open question arises now: could we use quantum computers to design and build better quantum computers? According to the nature of the problem, the answer should be yes, but to our best knowledge we do not know plans in this direction.

VIII. Other Open Issues: Industrial Challenges

QC might play a fundamental role in Industry, bringing innovative solutions to practical industrial applications. We briefly point some of the ones with more societal impact:

- *Automotive*. The industry of the automobile is moving fast towards electrical and hybrid technology. Batteries suffer from low autonomy and slow recharge. New batteries with higher energy density and small weight. QC offers higher computational power to investigate new chemical reactions and materials in this field [18].
- *Health – Drug discovery*. The formulation of candidates to medical compounds is very expensive in the Pharma industry. Simulating the behaviour of the proper molecules in compounds can save large amounts of money and make medicaments cheaper [19].
- *Automated planning & scheduling*. QA computers are specifically design for this type of optimisation problems, where QC seems to have a huge potential [18]
- *Quantum Artificial Intelligence*. Major challenges are: replace training by better QA, large datasets and QC, and standardised interfaces [20].

A very general SWOT analysis of the present status of QC is presented in Figure 5.

STRENGTHS	WEAKNESSES
• Theoretical framework • Quantum supremacy • Robustness of QC	• Complexity of QAs • Low number of qubits • Quantum software
OPPORTUNITIES	**THREATS**
• Large public investment • Main computer companies involved • Many industrial sectors interested	• Prize of services • Classical supercomputing easier to access • Difficult access to non experts

Fig. 5. A SWOT analysis on quantum computing.

IX. Conclusions

QC remains a challenge in itself. Much of the efforts in QC are targeting the hardware of quantum computers, rather than designing new QAs and quantum software to assist quantum programmers. The two QC paradigms described here point towards two different hardware architectures: QA designed for specific families of optimisation problems and quantum gate computers conceived as general-purpose quantum computers for a wider community of users. We can summarize the close future challenges of the two QC paradigms in the following items:

- Scalability with regards quantum decoherence, quantum errors and quantum fault tolerance to keep and improve QC robustness.
- Software testing and debugging tools for quantum programmers, with quantum platforms/interfaces for transparent use of quantum machines.
- Exhaustive research in new QAs for quantum gate computers and QUBO formulations for quantum annealers.

Acknowledgments

This research is partially funded by the Universidad de Málaga (DataPol UMA-CEIATECH-07), Consejería de Economía y Conocimiento de la Junta de Andaluía and FEDER under grant number UMA18-FEDERJA-003 (PRECOG); TAILOR ICT-48 Network (No 952215) funded by EU Horizon 2020 research and innovation programme.

References

[1] P.W. Shor, “Algorithms for quantum computation: Discrete logarithms and factoring”, 35th Annual Symposium on Foundations of Computer Science Proceedings, Santa Fe, USA, 1994
[2] L. Grover, “A fast quantum mechanical algorithm for database search, ArXiv quant-ph/9605043, 1996
[3] D.T. Gillespie, “A Quantum Mechanics Primer”, International Textbook Company, 1970
[4] A. Einstein, B. Podolsky, N. Rosen, “Can Quantum-Mechanical Description of Physical Reality Be Considered Complete”, Phys. Rev. 47, pp. 777, 1935
[5] D.P. DiVincenzo, “Quantum Computation”, Science 270, 255, 1995
[6] E. Knill, “Quantum Computing”, Nature, Vol. 463, 2010
[7] S. Kirkpatrick, C. D. Gelatt, Jr., and M. P. Vecchi, “Optimization by simulated annealing”, Science, Vol.220, pp.671, 1983
[8] B. Apolloni, N. Cesa-Bianchi, D. De Falco, “A numerical implementation of quantum annealing”, in Stochastic Processes, Physics and Geometry, Proceedings of the Ascona-Locarno Conference, July, 1988
[9] E. Cohen, B. Tamir, “Quantum annealing – foundations and frontiers”, Eur. Phys. J. Spec. Top. 224, pp.89110, 2015
[10] IBM website, https://quantum-computing.ibm.com/. Accessed on: June 2021
[11] D-Wave Systems website: https://quantum-computing.ibm.com/. Accessed on: June 2021
[12] E. Boros and A. Gruber, “On Quadratization of Pseudo-Boolean Functions”, International Symposium on AI and Mathematics, 2012
[13] P.W. Shor, “Scheme for reducing decoherence in quantum computer memory’ ’, Phys. Rev. A, 52, 1995
[14] A.M. Steane, “Error Correcting Codes in Quantum Theory”, Phys. Rev. Lett. 77, 793, 1996
[15] A.M Steane, “Tutorial on Quantum Error Correction”, in “Quantum Computers,Algorithms and Chaos”, Proceedings of the International School of Physics Enrico Fermi, pp.132, IOS Press, 2006
[16] D. Aharonov, M. Ben-Or, “Fault-tolerant quantum computation with constant error”, Proceedings STOC'97, pp. 176-188, 1997
[17] M.A. Nielsen, I.L. Chuang, “Quantum Computation and Quantum Information”, Cambridge University Press, 2010
[18] A. Luckow, J. Klepsch, J. Pichlmeier, “Quantum Computing: Towards Industry Reference Problems”, Digitale Welt, vol.5, num.2, 2021
[19] Y. Cao, J. Romero, A. Aspuru-Guzik, “Potential of quantum computing for drug discovery”, IBM Journal of Research and Development, vol. 62, no. 6, 2018
[20] T. Gabor, L. Sünkel, F. Ritz et al., arXiv:quant-ph/2004.14035, 2004

XIX Conferencia de la Asociación Española para la Inteligencia Artificial

I Taller sobre Modelos Generativos Profundos Aplicados (AGM)



GAN-based data augmentation for crack detection

Alberto Botana López*, Daniel Gordo Martín,
Adrián Alonso Rial, Jacobo Otero Tranchero
and Santiago Muiños-Landín*
Technological Centre AIMEN, C/ Relva, 27 A. Torneiros - 36410 Porriño - Pontevedra, Spain
*alberto.botana, santiago.muinos@aimen.es

***Abstract*—The application of Machine Learning for manufacturing has become a reality with the ongoing digitalization of factories. Generative Models are a promising but not yet exploited enough technology in manufacturing frameworks. In this work, we apply Generative Models for a quality inspection problem in order to generate synthetic data for training. We show how through the use of a Generative Adversarial Network, synthetic images are generated to augment the training data set of a crack detection system. We show the impact of such data generation in the detection ratios of our system and how for problems with inhomogeneous defect typologies distribution, Generative Models can provide a solution for artificial defect generation to enhance detection capabilities based on neural networks.**

I. Introduction

Manufacturing Frameworks are a perfect scenario for the deployment of Machine Learning (ML) techniques [1], the validation of their applicability and also the incubation of new algorithms and ideas. The huge amount of data that comes out from different widely sensorized manufacturing frameworks and its current digitalization has triggered the evolution of several technologies towards the autonomy, adaptability and self-organization of devices deployed along factories all around the world. However, one bottleneck for the efficient application of some ML techniques that rely in the use of Artificial Neural Networks (ANN) is the large amount of data required for training. In particular, many Quality Inspection procedures are based in Convolutional Neural Networks (CNN) [2] for image processing that need large amount of images for it. This, added to the obvious fact that in general factories try to minimize the existence of defects, makes that having large amounts of images of defects tends to be difficult. In addition, defects usually appear in a way that can be classified by some common features meaning that different defect typologies can be identified for the same process and the frequency at which the different typologies appear is usually different. Which means that examples of some defect typology can be massively acquired and examples of some other can be rarely seen. The impact of this, makes that a quality control procedure might show really good results for some typology while remaining almost blind for the detection of another.

In this work we take as initial point a CNN-based crack detection system developed in previous own work. Such system detects cracks from metallic parts within a manufacturing framework in the automotion sector. The detection is based on the non-destructive use of magnetic nanoparticles [3] to highlight the regions where cracks appear. The defects of our scenario are distributed in different typologies and detection ratios are quite different depending on them. We use a generative approach to generate synthetic image data for training of our CNN with a significant impact on the detection ratios. This section gives a short overview of the non-destructive testing technique used in our context and also of the state of the art Generative approach followed for our purpose. The next section introduces the problem and the different elements involved in its solution which is explained in the third section of this paper. Finally results are shown and commented and conclusions are exposed with a description of future work triggered by the presented development.

A. Non-Destructive Testing Method: Magnetic Particles

The automation of crack detection, has been topic of different developments within last years [4] in order to achieve similar or higher accuracy compared with traditional methods while making at the same time jobs more attractive for workers. Until now, this method, when based on the use of magnetic particles [5], consisted on a factory worker spending full working days inspecting these metallic pieces under black light looking for clusterings of these kind of fluorescent particles. This technique, known as *magnetic particle inspection* [6] belongs to the so called non-destructive testing techniques [7]. In this case, by magnetizing a ferromagnetic material by direct or indirect magnetization and applying a coat of ferrous particles mixed with fluorescent dye, these particles result attracted towards the cracks or corrosion pits on the surface, since these defects create a flux leakage field in the magnetized material. These clustered particles can then be easily seen when shone upon with black light, since it interacts with the fluorescent dye making it shine on a bright green color (see Fig.1).

B. Generative Adversarial Networks (GAN)

Generative Adversarial Networks, or GANs are a class of machine learning frameworks designed by Goodfellow et al. [8] in 2014. GANs have been used extensively for a wide range of tasks such as creating faces [9], domain-changing

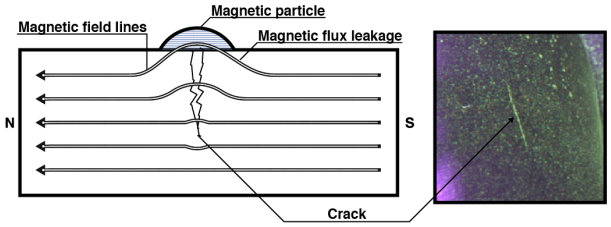


Fig. 1. Magnetic field inside a ferromagnetic material with a crack. On top of it rests a high concentration of magnetic particles (blue volume). The zoom of the image shows one example of the cracks detected in our case by the accumulation of these particles (bright green line) showing fluorescence response under the application of UV light.

image-to-image translation [10] and even video synthesis [11]. These models consist of two separate sub-models: A *generator* that tries to come up with new images from a dataset, and a *discriminator* that, having access to the training database, tries to differentiate an original picture from the database from a picture created by the model's generator. These two sub-models compete with each other in the form of a zero-sum game.

In this training process, the generator will start by generating random noise in a latent space, which will be transformed into an image by running it through several transposed convolution layers which will upscale the image according to its weights up to a fixed size. Once that has been done, the discriminator, by running the generator output through a series of convolutional layers, will try to guess whether the image produced was indeed forged by the generator or it belongs to the real dataset. This way, the objective of the generator is to maximize the error rate of the discriminator by effectively fooling it, since that means that its outputs are so good that they can't be distinguished from the real data. In order for both of the sub-models to be trained properly, independent backpropagation is applied to both parts. A schematic diagram representing this process can be found on Fig. 2.

II. Problem

For the problem addressed in this work, we analyzed metallic parts in the production line and prototypes of a factory in the automotion sector. The main source of defects in these pieces is the appearance of cracks. A crack is defined as a thin, irregular line over the surface of a piece that is the beginning of a future breakage. Therefore, their early detection becomes critical. They are produced by over-heating or variations in the process parameters during the fabrication. The manufacturing method is based on batch production, where items with similar characteristics are produced in the same production run. We developed a clustering algorithm based on *TSNE* [12] and *K-Means* [13] [14] that helped us to identify up to 6 different classes of crack based on its morphology and location within the piece. An example for every crack type can be found on Fig. 3. The main problem we addressed and where this work is focused, was to achieve similar detection rates for all the defect

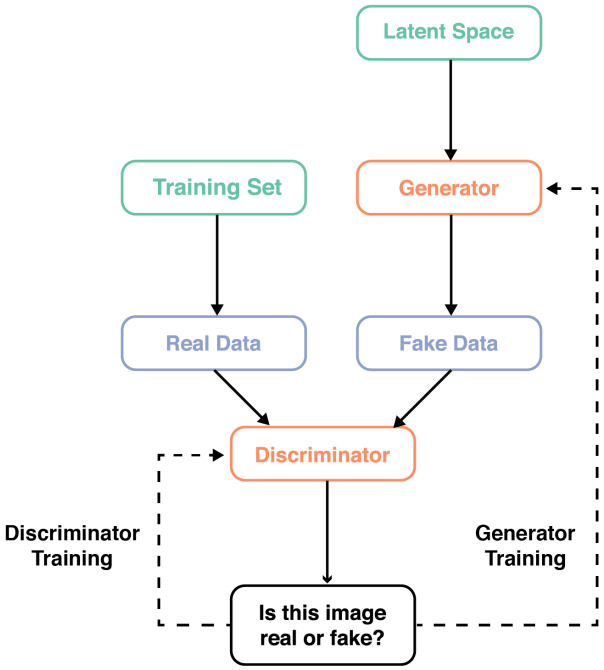


Fig. 2. Schematic rundown of a GAN training process. Two Networks, Generator and Discriminator (both in orange), in a game theory context using real data to generate synthetic images.

typologies, given that such asymmetry in the defect typology distribution and specially the low number of examples for some typologies, made the accuracy of the detection procedure poor some certain typologies while being high for others. To automate the crack detection process we developed an image acquisition system to take images from the piece to inspect and a neural network responsible for classification. A detailed definition of this detection system is out of the scope of this work, although we will remark some details in next sections.

A. Detection Setup and Dataset

To create the dataset, a custom setup was designed to take images of the pieces. It was formed by two fixed cameras, an ultraviolet light lamp and an industrial claw. The two cameras were the model acA2000-50gc [15] from Basler, which are industrial cameras that can deliver up to 50 frames per second at a maximum of 2 MP resolution. The process was the following: one of the faulty pieces was placed in the claw and it rotates every second from 20 to 30 degrees according to the size of the piece. In the moment the piece was motionless, the two cameras took an image from different angles. So we were able to obtain between 12 and 16 images for every piece. These images were stored in HDF5 [16] files for its subsequent labelling. For manual labelling an interface was programmed

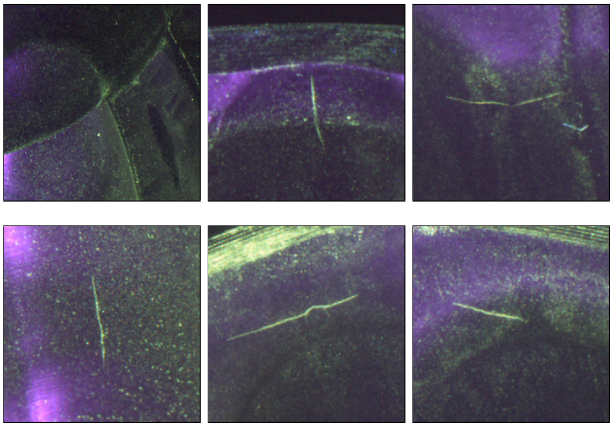


Fig. 3. Examples of each of the six crack types identified by clustering. The typology depends on the form of the crack but also on its position within the piece analysed.

using python. The information flow of this data acquisition process can be seen in Fig. 4.

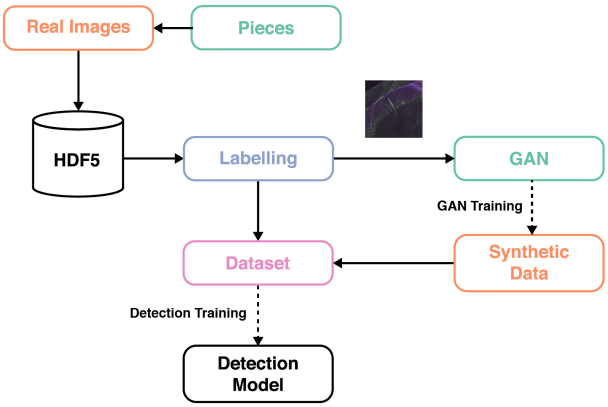


Fig. 4. Information flow of the detector system. The cracks from the pieces are digitalized taking images from them. All the images are included in a dataset in HDF5 format where they are manually labelled. The GAN uses real images for training to generate synthetic ones that are included in the dataset to train the detection model.

For industrial companies the cost to serve nonconformities is very high, so defects tend to be a rare event, in the order of PPM (parts per million).Therefore, the dataset we worked on was characterized by the low number of cracks examples available. We also found problematic the lack of homogeneity in the classes distribution, so our detector performed better for the majority class than for the rest. We can see the class distribution in Fig. 5. Furthermore, defective pieces from the same production batch tend to be quite similar, as they have suffered the same production conditions, so many of their cracks did not add additional variety to the dataset.

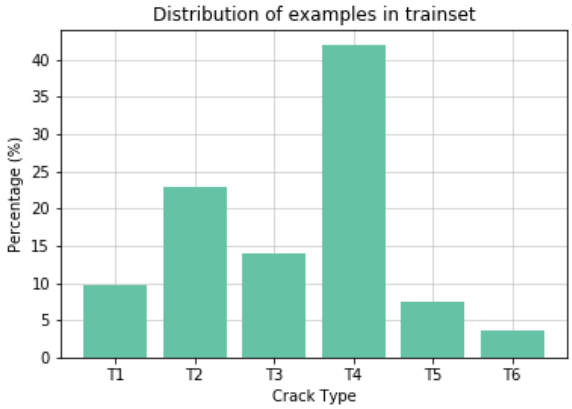


Fig. 5. Cracks type distribution. The different frequencies at which different typologies of defects are detected results in a highly asymmetric dataset where some typologies can only be poorly trained for detection.

B. Detection Network

The detection method we used was based on a fully convolutional network (FCN) [17]. We chose a ResNet50 [18] network with pre-trained weights as a feature extractor. The last fully connected layer of this network was removed and replaced by a block formed by two consecutive layers:

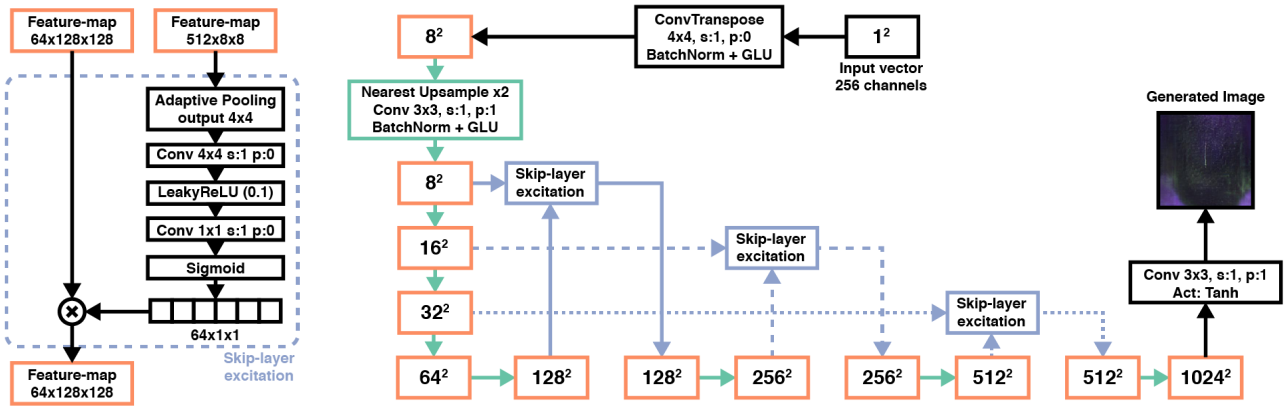
- Average pooling layer with a pool size 7x7 and stride 1.
- Convolutional 1x1 with stride of 1 and zero padding.

Thus, the ResNet50 outputs a feature map, showing the spatial location of features in the image. This volume is fed to the new block which outputs a segmentation map. Finally, this segmentation map is overlapped to the whole image and region of interest are extracted with skimage measure module [19]. Some details of the experimentation process:

- **Optimization** We used SGD with a learning rate of 10^{-3}, a momentum of 0.9 and a weight decay of $5 \cdot 10^{-4}$. We noticed that this method outperformed other optimizers for this problem, as it had a faster convergence. We did not freeze any layer as we got a better performance training the whole network.
- **Patch-sampling** In a first stage of the project, background samples were chosen at random. In a later stage, we used trained models to select the most interesting background regions and include them in the training.
- **Augmentation** To increase the size of the training set we applied random data augmentations during training. The transformation applied were shear mapping, image zooming, horizontal and vertical shift, brightness, rotation and flipping.

III. Proposed Solution

The proposed solution to fix this lack of training data was to use **FastGAN**, a minimal and lightweight GAN proposed by Liu et al. [20]. This proposed model was centered around the



(a) Skip-Layer Excitation unit

(b) Generator workflow

Fig. 6. Schematic of the FastGAN generator architecture.

premise of being able to produce high fidelity results without high-end equipment, training only for a few hours on a single NVIDIA RTX-2080 GPU unit. The architecture of this model can be seen on Fig. 6 and 7, both reproduced from the original authors in [20].

A. Generator

In order to be able to generate high-fidelity results, the generator needs a lot of up-sampling layers, since each of these will effectively double the resolution of the output. However, a deeper model with more convolution layers will end in a higher computational cost and a longer training time. To overcome this, He et. al [18] designed the residual block structure, which connects the layers using a skip-layer connection, which link two layers further than one block apart, effectively skipping one or more layers and thus boosting the gradient signal. This method, however, while decreasing the raw amount of time required, it does increase its computational cost.

To compensate for this complexity increase, the authors reformulated this skip-layer idea, coming up with the Skip-Layer Extraction (SLE) module. This module firstly implements skip-connection as an channel-wise multiplication of the activation layers from different convolutional layers. This differs from the ResBlock, as in that method implements skip-connection as an element-wise addition instead of the multiplication. In addition, they use a different approach regarding the sample resolution of the skip-connections. Previous authors only used skip-connections when dealing with the same resolution, while the authors perform skip-connections between resolutions with a much broader range, like 8^2 and 128^2. This is possible since, by implementing the skip-connections as a multiplication instead of an addition, the spatial resolution is no longer needed. These two main advantages allow for a much simpler computation, affording both in complexity and training time.

B. Discriminator

In order to provide the discriminator with a strong regularization, it was treated as an encoder as a whole and trained it with small decoders, forcing the discriminator to extract image features. These decoders are optimized along with the discriminator using a reconstruction loss following:

$$\mathcal{L}_{\text{recons}} = \mathbb{E}_{\mathbf{f}\sim D_{\text{encode}}(x), x\sim I_{\text{real}}}\left[\|\mathcal{G}(\mathbf{f}) - \mathcal{T}(x)\|\right] \quad (1)$$

Where $\mathbf{f}$ is the intermediate feature-maps from the discriminator, the $\mathcal{G}$ function contains the processing on $\mathbf{f}$ and $\mathcal{T}$ represents the processing on sample x from real images I_{real}.

This discriminator model, shown in Fig. 7 employs encoders on only two scales: $\mathbf{f}_1$ on 16^2 and $\mathbf{f}_2$ on 8^2, along with only four convolution layers in order to produce images with a resolution of 128x128 pixels. By acting this way, the computational cost of the regularization is reduced by much more than other regularization methods. In order to obtain the final image, $\mathbf{f}_1$ is cropped with $\frac{1}{8}$ of its original size, then the real image is cropped by the same amount, thus obtaining I_{part}. Then the decoders produce I'_{part} from the firstly cropped $\mathbf{f}_1$ and I' from $\mathbf{f}_2$. Lastly, the discriminator and decoders are trained together to minimize the loss depicted on equation 1 by trying to match I' to I and I'_{part} to I_{part}.

Taking this reconstructive approach allows the discriminator to obtain a more comprehensive representation from the inputs, since it's able to cover the big picture through $\mathbf{f}_2$ as well as the finer details through $\mathbf{f}_1$.

IV. Results

After training the model for 150 epochs, it was prompted to generate a batch of 3000 images. Some of those generated images can be seen on Fig. 8. On most cases, the produced result mimic the original images successfuly, although there are some examples, like those shown on figure 9 which don't represent any kind of crack contained within the original

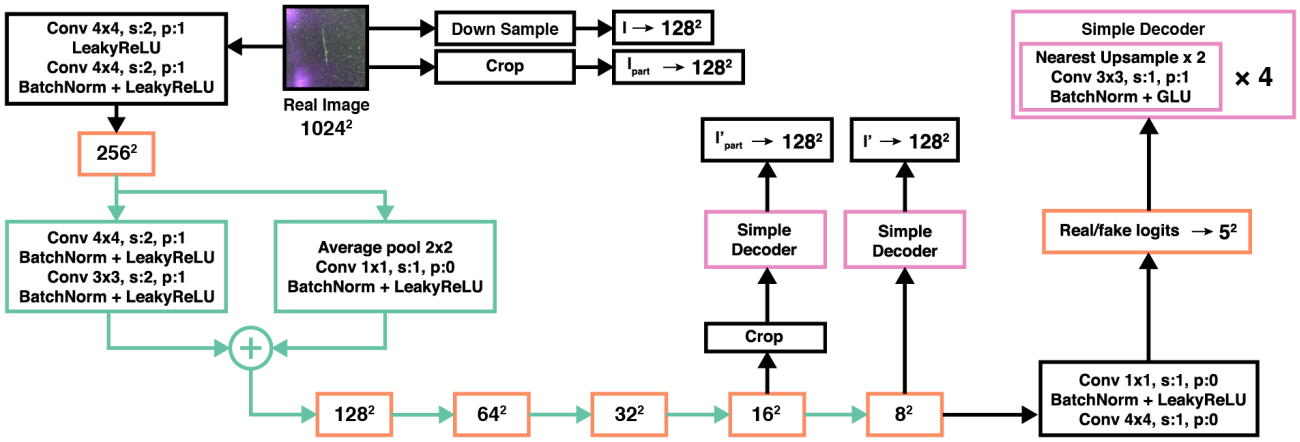


Fig. 7. Schematic of the FastGAN discriminator architecture.

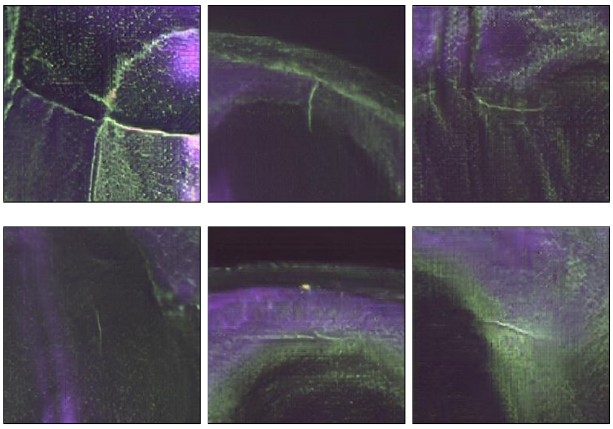


Fig. 8. Examples of Generated images outputted by the model for each of the six different crack typologies. The GAN managed to generate images even for those typologies that where clearly minority.

dataset. These images, as is often common when training GANs, represent complete nonsense while trying to keep a familiar style with the dataset. These faulty examples will be taken care of in future models by implementing a reinforced model to this GAN architecture, effectively building an Object-Reinforced Generative Adversarial Model, or ORGAN [21]. This generated data was added to our train set and a new detection model was trained. We compared the new detector performance with other detection model, using the same architecture, to which these images were not added to the training set. Both models were trained using the same parameters, some of them were defined above. The test set consisted in 239 pieces from various production batches never seen by the system before in order to avoid any bias. Every piece is represented by between 12 and 16 images. The test set had 118 defective pieces and 121 non-defective ones. A defective piece was considered to be correctly classified if we detect, at least, a crack in one of the images. Finally, the metric selected

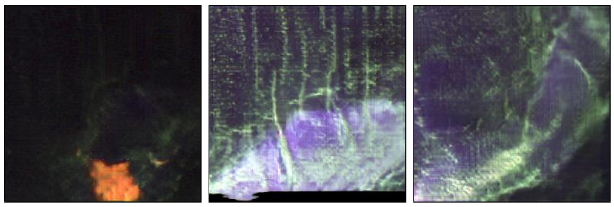


Fig. 9. Flawed images generated by the model. The figure shows examples of some images that appear within the synthetic dataset and need to be manually filtered.

for comparison was f1-score. Thus, the model trained with synthetic data was able to classify correctly the 95.7% of the defective pieces and the f1-score metric was 0.83, meanwhile the model trained without the synthetic data classified correctly the 90% of defective pieces and the f1-score was 0.79.

A. *Future Steps*

Our main purpose in order to perfect this model is to enhance this architecture and turn it into an ORGAN, so that it can be fine tuned and give access to more control over the crack generation process. This way, the faulty crack generation problem would be mostly nonexistent and we would be able to produce a specific type of crack according to the model's needs. Additionally, in order to increase model accuracy, a bigger crack dataset will be used on further model trainings.

V. Conclusion

In this paper, an application of Generative Adversarial Models has been proposed in order to produce synthetic data to increase the size of a dataset and, thus, improve the performance of a FCN-based detector. We have shown that GAN generated synthetic data can be a useful tool for problems where we have to deal with small datasets. We consider this method can complement and overcome the limitations of traditional data augmentation techniques which only focus on 2D transformations.

VI. Aknowledgments

AB, DG and SM want to thank the fruitful comments and discussions of all the members of the Artificial Intelligence and Data Analytics Lab at the Smart Systems and Smart Manufacturing Department of the AIMEN Technology Centre.

References

[1] Z. Kang, C. Catal, and B. Tekinerdogan, "Machine learning applications in production lines: A systematic literature review," *Computers Industrial Engineering*, vol. 149, p. 106773, 2020.

[2] A. Dhillon and G. Verma, "Convolutional neural network: a review of models, methodologies and applications to object detection," *Prog Artif Intell*, vol. 9, p. 85–112, 2020.

[3] P. Prabhu Gaunkar, D. C. Jiles, and G. V. Prabhu Gaunkar, "Detection of surface cracks in ferromagnetic materials by c-scan mapping of residual stresses using barkhausen emissions," *AIP Advances*, vol. 10, p. 015246, 2020.

[4] W. Zhang, Z. Zhang, D. Qi, and Y. Liu, "Automatic crack detection and classification method for subway tunnel safety monitoring," *Sensors*, vol. 14, pp. 19 307–19 328, 2014.

[5] A. P. Mouritz, "Introduction to aerospace materials. nondestructive inspection and structural health monitoring of aerospace materials," *Woodhead Publishing*, vol. 10, pp. 534–557, 2012.

[6] H. Lester, "Magnetic particle inspection," in *Symposium on Magnetic Particle Testing*. ASTM International, 1945.

[7] M. MR Jolly, A. Prabhakar, B. Sturzu, K. Hollstein, R. Singh, S. Thomas, P. Foote, and A. Shaw, "Review of non-destructive testing (ndt) techniques and their applicability to thick walled composites," *Procedia CIRP*, vol. 38, pp. 129–136, 2015.

[8] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *arXiv preprint arXiv:1406.2661*, 2014.

[9] S. . A. T. Karras, T.and Laine, "A style-based generator architecture for generative adversarial networks," *In arXiv preprint arXiv:1812.04948*, p. http://arxiv.org/abs/1812.04948, 2019.

[10] J. A. Grant-Jacob, B. S. Mackay, B. J. A. G., Y. Xie, and M. Heath, D.J.and Loxham, "A neural lens for super-resolution biological imaging," *Journal of Physics Communications*, vol. 3, no. 6, p. 065004, 2019.

[11] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, G. Liu, J. Tao, A.and Kautz, and B. Catanzaro, "A neural lens for super-resolution biological imaging."

[12] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, 2008. [Online]. Available: http://www.jmlr.org/papers/v9/vandermaaten08a.html

[13] H. Steinhaus, "Sur la division des corps matériels en parties," *Bull. Acad. Polon. Sci*, vol. 1, no. 804, p. 801, 1956.

[14] S. Lloyd, "Least squares quantization in pcm," *IEEE transactions on information theory*, vol. 28, no. 2, pp. 129–137, 1982.

[15] Basler AG. (2021) Basler ace aca2000-50gc. [Online]. Available: https://www.baslerweb.com/en/products/cameras/area-scan-cameras/ace/aca2000-50gc/

[16] The HDF Group. (2000-2010) Hierarchical data format version 5. [Online]. Available: http://www.hdfgroup.org/HDF5

[17] T. D. Jonathan Long, Evan Shelhamer, "Fully convolutional networks for semantic segmentation," *arXiv:1411.4038*, 2014.

[18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[19] S. Van der Walt, J. L. Schönberger, J. Nunez-Iglesias, F. Boulogne, J. D. Warner, N. Yager, E. Gouillart, and T. Yu, "scikit-image: image processing in python," *PeerJ*, vol. 2, p. e453, 2014.

[20] B. Liu, Y. Zhu, K. Song, and A. Elgammal, "Towards faster and stabilized {gan} training for high-fidelity few-shot image synthesis," in *International Conference on Learning Representations*, 2021. [Online]. Available: https://openreview.net/forum?id=1Fqg133qRaI

[21] G. L. Guimaraes, B. Sanchez-Lengeling, C. Outeiral, P. L. C. Farias, and A. Aspuru-Guzik, "Objective-reinforced generative adversarial networks (organ) for sequence generation models," *arXiv preprint arXiv:1705.10843*, 2017.

GANs for Urban Design: Learning Subtle Morphological Differences.

1st Stanislava Fedorova
AIUC School
Politecnico di Milano
Milan, Italy
stanislava.fedorova@mail.polimi.it

***Abstract*—The use of Generative Adversarial Networks (GANs) in generative urban design is currently underexplored. This research addresses this issue by expanding the knowledge of the possible practical use cases of this instrument in the urban design field by exploring the model's ability to learn small morphological details of the urban fabrics that determine the identities of the urban zones and play an essential role in differentiating between the cities with similar architectural background. Moreover, this work highlights the potential uses of the instrument in the industries, that are possible due to the ability to learn these small details from the urban surroundings.**

***Index Terms*—generative adversarial networks, generative design, deep learning, artificial intelligence, urban design**

I. Introduction

From the introduction of the concept of Generative Adversarial Networks (GANs) [1] there have been multiple attempts to introduce it to various fields. There is massive amount of research on the application of GANs to finance, commerce, physics, biology, medicine and others. At the same time, the studies on the practical application of GANs in urban design and architecture are quite limited and theoretical research does not imply any practical implementations of the technology.

However, architecture and urban design are exactly the fields where the advantages of the artificial intelligence, in particular, of the generative networks would find the most use cases, due to their capability of learning visual features in a self-supervised manner. In urban design there are multiple interdisciplinary parameters to be considered when making a project. The variety and the amount of the features that is often a complication for the planner, can potentially not be such for a neural network designed to deal with many parameters.

In this work we explore the possibility of the application of GANs to the urban design problem from the morphological point of view, in order to see its ability to provide compliant solutions in various contexts. We would also like to discuss the potential developments and possibilities of the practical implementation of the proposed tool.

II. Related Works

GANs have a limited use in the area of urban design and architecture. Since their introduction [1], though, there has been a number of works involving the usage of this algorithm; most, however, refer to the analysis stage [9] [10] or the visualization part of a project [13]. The majority of these works rely on genetic algorithms [11] or semi-automatic approaches [12] to generative urban design.

A project "This map does not exist" [8] does address the creative stage of an urban design project; it uses StyleGAN2 [7] in order to generate images simulating OpenStreetMap image samples. The practical use of this tool is limited, as it produces the urban design solutions that are not conditioned by the existing environment.

This research continues the work [3] that uses Image-to-image translation neural networks [2] in order to generate urban block design based on the existing urban surroundings. However, the differences in the chosen morphologies are rather defined

in their distinguishing characteristics, which renders it easy to reproduce a solution closer to a certain 'class'. The present work explores the potential use of the GAN algorithm in the similar urban contexts.

III. Datasets

All the four datasets used in the experiments were composed using the same data acquisition principle. Datasets refer to the following cities:

- Milan
- Turin
- Florence
- Bologna

TABLE I
AMOUNT OF SAMPLES PER EACH OF THE DATASETS.

City	*Number of train samples*
Milan	1193
Florence	503
Bologna	784
Turin	1115

The datasets consist of structured data (images) in RGB colorspace, as in Fig. 1. The images are diagrammatic representations of the parts of a city masterplan and include the following elements: building units, road lines and hierarchy, railways.

The data were sampled taking into consideration the heterogeneity of the city (empty blocks, blocks with atypical functions and blocks with morphological characteristics too different from the average were not included into the datasets; the datasets were balanced based on the blocks' functions). Cities are products of a long-term design process involving multiple actors and influences, which renders them heterogeneous, thus largely increasing the variance and the bias in the dataset, as some parts of the city are vaster and more homogeneous than the others. Due to this reason, the data were sampled from the limited areas of the cities with consistent urban fabric characteristics.

Every dataset consists of two image sets. The input set is fed into the generator model, while the target set is fed into the discriminator model as the desired output. The difference between the two

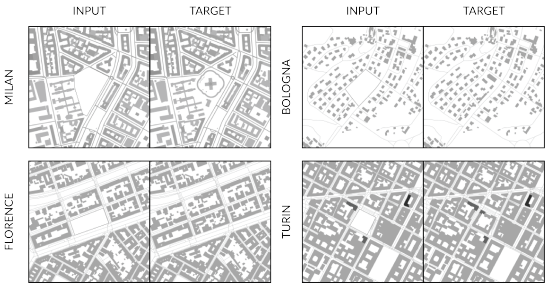


Fig. 1. Sample from the four datasets of urban blocks: the first column shows the samples from the input set, the second column shows the samples from the target set.

sets in our case is that in the input set, the block of interest is eliminated and replaced with a white polygon with thin black border of 1px. This block of interest can be seen as a 'construction site', as the model needs to produce an urban design solution on this site, as can be seen in Fig. 1.

Each dataset was produced using QGIS software [7] and python scripts, made available for the public [6]. All the images were produced in a consistent scale 1:3000; image dimensions are 256x256.

The data were taken from the Openstreetmap.org [14] available under the Open Database Licence. No modifications have been made to the data provided.

IV. Architecture

Typically, a GAN consists of two competing neural networks, as denoted in the name of the algorithm; first one, responsible for the content generation, is called Generator, while the second is called Discriminator and is responsible for the content evaluation based on the comparison with the existing samples provided to it. As this research is focused on the design aspect and the capture of small details of the urban environment, it was decided to use the architecture for Image-to-Image translation [2]. Among the advantages of the chosen model are the Unet-based architecture of Generator [15] and a PatchGAN-based Discriminator [2]. These two factors allow for a more detailed design generation as well as for the successful learning of the small details by both the networks due to the PatchGAN's structure that evaluates every 70x70

patch of the image separately to estimate the overall quality of the generated content. The architecture of the chosen model is reported in Fig. 2 with the amount of layers, and the input size for each layer.

The implementation of the chosen architecture was Pytorch-based [4] [5] with the use of the adaptive learning rate based on comparison with the previous accumulated losses, starting from learning rate equal to 0.0001. The loss function for the discriminator was chosen to be binary cross entropy between target and output logits. The generator's loss function is the difference between L1 loss function and discriminator's loss. The models were trained with the application of Adam optimizer.

The models have been trained on Google Colab platform with a single NVIDIA Tesla K80 GPU. Each training took approximately 30 minutes.

V. Experiments

Having followed the existing research in the field of generative urban design with the use of GANs [3], several cities were chosen for the experiments. Previously, GANs morphology learning and generation capabilities were tested in the contexts with completely different morphology where the variations of urban fabric were clearly visible. In our research the attention is focused on the cities coming from a similar architectural background with more subtle urban morphology differences.

There were trained four GAN models for each of the chosen cities. Afterwards, the models were tested on the familiar urban surroundings from the test set as well as on the other cities' urban contexts. The results are reported in Fig. 4.

It was noted that, while performing well on the familiar urban context, models with a limited (due to the city size) amount of samples, failed on an unseen urban context, as in the model of Bologna in Figure 4 that did not generate a sufficient design in Milan, Turin and Florence, while producing a coherent solution on a Bologna test sample.

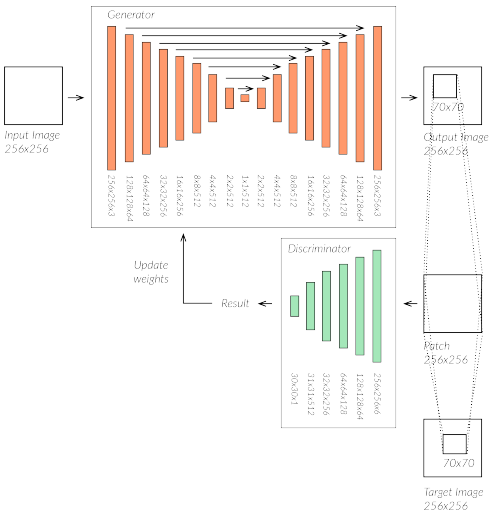


Fig. 2. High-level representation of Pix2pix architecture, as in [2]. U-net based generator and PatchGAN discriminator.

VI. Evaluation

Due to the difficulties in obtaining qualitative evaluation of the produced samples this work uses a convolutional neural network (CNN) in order to estimate samples' similarity with respect to the typical morphology of the cities. First, the network (ResNet50 [16]) was trained on the dataset with the images of the blocks of each of the cities; a sample of such an image can be seen in Fig. 3. This dataset was split into train and validation sets with 15 percent of validation images. One city is one class. Then it was tested on the generated blocks for each of the four cities; there were 65 test images per class. The metric used for the evaluation of the network's performance was accuracy.

From the Table II it can be observed that the samples are not produced with the same precision as in the research on completely different morphologies [3] and the classification model does not recognize some of the samples from the generated set. It can be clearly seen on the example of Bologna: since on quite a few of the samples the model failed to

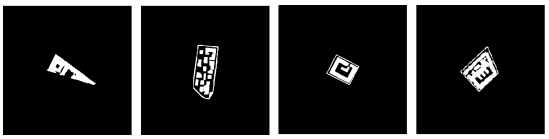


Fig. 3. Samples from the classification dataset used for the evaluation. Left to right: Florence, Bologna, Turin, Milan.

produce a solution, these images were not classified correctly. However, in the cities with the sufficient amount of samples, the model classified correctly produced design solution from the morphological point of view. Also, as in the case with international style translation, the model adapts the solution to the surrounding environment aligning the volumes to the adjacent blocks.

It should be noticed that, while between different countries the shape and the size of an urban block can be a distinguishing feature itself for the classifier, in the case of Italian cities coming from the same historical region and this from the same background, this bias is more limited.

TABLE II
CLASSIFICATION ACCURACY FOR VAL AND TEST SAMPLES.

City	Valid Acc.	Test Acc.
Bologna	0.87	0.21
Florence	0.94	0.68
Milan	0.87	0.66
Turin	0.83	0.8
Total	0.87	0.59

VII. LIMITATIONS

There is a number of limitations regarding the suggested approach. Some of them refer to the particular experimental process while the others refer to the chosen architecture.

Limitations referring to the experimental setup:

- **Urban scale** All the images in the train and the test sets were produced in the same scale, so the model is not able to differentiate between different urban scales. And thus, will not produce coherent solutions for the images in a scale different from the one of the training;
- **Number of blocks designed** All the images in the train and the test set had only one block of interest located approximately in the center of the image. As can be seen in Fig. 5 this prevents the model from designing more than one block per image. It does recognize the blocks of interest though, by generating some noise on the borders of these blocks.

Limitations referring to the architecture:

- **Image clarity** Images lack clarity and precision of the contours compared to the traditional urban designs and vector images. This limitation can potentially be resolved by the data augmentation and image post-processing.
- **Creativity** The design solutions produced are based on the existing urban context of the city they were trained on. The lack of novelty renders the tool unsuitable for landmarks design.
- **Dataset size** The models are not capable of producing a coherent solution in an environment different from the familiar one if the train set is limited. This limitation can potentially be resolved by the dataset expansion.

VIII. OPEN PERSPECTIVES

This research has extended the existing research in the field of generative design [3]. The future developments could explore, for instance, the current limitations related to the experimental setup can be addressed in order to see the model's ability to learn the morphological characteristics on different scales as well as the scales themselves or the correct amount of blocks to generate. It would also be interesting to train a single GAN model on various urban contexts. Another possible development could be the integration of the further conditions into the generation process apart from morphology, e.g. urban density, green area proportion, heat emissions. Moreover, it could be interesting to add the third dimension to the generated urban design projects; this is a promising study direction due to the latest development of 3D Deep Learning frameworks.

IX. PRACTICAL USE

The use of the image-to-image translation 'as-is' in the creative urban design process is currently difficult to imagine due to the mentioned limitations of the model, especially due to the constraint related to creativity. At the same time, there is a number of practical applications that can exploit the model in its current state. The tool can be used for the simulation of the urban environment based on certain conditions when developing a masterplan, e.g.



Fig. 4. Test set results. Rows represent the existing urban context of the city, columns represent the cities the models were trained on (the expected morphology of the urban block). E.g. cell [3, 4], row 3 column 4 represents the Turin model tested within Bologna urban context: Turin urban block design in Bologna. The diagonal cells represent the models trained and tested on the same city.

simulating the urban morphology based on the defined functional zoning; another use case is the use of the tool in the didactic scopes, as an insight for the students of architecture and urban design into the relationships between the urban volumes and spaces. These two use cases could be doubtful when comparing the cities with the completely different morphologies; our research, however, observes that the ability to learn distinguishing characteristics of the highly related urban contexts, which allows for the instrument to be used in this way.

Moreover, the model can be used in gaming industry in order to simulate an urban fabric while not reproducing its exact details. It can be especially useful for the procedural environment generation.

X. Contribution

This paper aims to contribute to the existing research of the use of GANs in the field of generative



Fig. 5. An experiment showing generation of several block designs for one input image in the cities of Milan and Bologna. The columns are: the input image, output 1 with one block design, output 2 with 2 block designs required, output 3 with 3 block designs required; target image. The blocks of interest are highlighted with color (1 - red, 2 - green, 3 - blue) in order of their disappearance in the experiments for illustrative purposes.

urban design by exploring the results of the model's learning process in the contexts with subtle urban differences. In the research it was noticed that the model is able to learn small morphological details contributing to the zone's identity.

XI. Conclusion

It can be observed that GANs are not only capable of producing an urban design solution compliant with the urban surroundings, but also learn small differences between similar urban morphologies and reproduce them as small details, slightly modifying the design. The research conducted in this paper suggests that this tool has the potential to be used in certain practical contexts in urban design, as well as in gaming industry, mostly due to its advantages: relatively low time and effort consumption, ability to learn independently from the visual features.

Acknowledgment

I would like to express my gratitude to my thesis supervisors, Prof. Luigi Mussio and Prof. Federico Deambrosis and Prof. Andrea Bonarini for their support and encouragement during this research.

References

[1] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley,Sherjil Ozair, Aaron Courville, and Yoshua Bengio "Generative adversarial nets. In advances in neural information processing systems", Z. Ghahramani,M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger (Eds.), Vol. 27. CurranAssociates, Inc., 2672–2680. 2014.

[2] P. Isola, J. Zhu, T. Zhou, and A. A. Efros "Image-to-image translation with conditional adversarial networks". In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 5967–5976. 2017.

[3] S. Fedorova, "GANs for urban design", In proceedings of 12th Symposium on Simulation for Architecture and Urban Design (SimAUD 2021), April 2021, in press.

[4] Deeplearning.ai "GANs specialization" unpublished. 2020. https://www.deeplearning.ai/generative-adversarial-networks-specialization/

[5] S. Fedorova "Github repository with published models." unpublished. https://github.com/STASYA00/UrbanGen. 2020

[6] S. Fedorova "Github repository for urban datasets generation." https://github.com/STASYA00/urban_datasets unpublished. 2020.

[7] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, T. Aila "Analyzing and Improving the Image Quality of StyleGAN". Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020, pp. 8110-8119

[8] I. Boates "This map does not exist project." unpublished. 2020. https://github.com/iboates/thismapdoesnotexist

[9] K. Steinfeld "GAN loci". ACADIA 19: Ubiquity and Autonomy, Proceedings of the 39th Annual Conference of the Association for Computer Aided Design in Architecture (ACADIA), The University of Texas at Austin School of Architecture, Austin, Texas 21-26 October. 2019. pp. 392-403

[10] C. Waibel, R. Zhang, T. Wortmann "Physics Meets Machine Learning: Coupling FFD with Regression Models for Wind Pressure Prediction on High-Rise Facades", In proceedings of 12th Symposium on Simulation for Architecture and Urban Design (SimAUD 2021), April 2021, in press.

[11] D. Nagy, L. Villaggi, and D. Benjamin "Generative urban design: integrating financial and energy goals for automated neighborhood Layout", In proceedings of 9th Symposium on Simulation for Architecture and Urban Design (SimAUD 2018). 2018.

[12] I. Osintseva, R. Koenig, A. Berst, M. Bielik, S. Schneider "Automated parametric building volume generation: a case study for urban blocks", In proceedings of 11th Symposium on Simulation for Architecture and Urban Design (SimAUD 2020), 2020.

[13] A. Noyman, K. Larson, "A deep image of the city: generative urban-design visualization", In proceedings of 11th Symposium on Simulation for Architecture and Urban Design (SimAUD 2020), 2020.

[14] OpenStreetMap contributors, "OpenStreetMap", Data retrieved from "https://www.openstreetmap.org", 2017.

[15] O. Ronneberger, P. Fischer, T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation", In: Navab N., Hornegger J., Wells W., Frangi A. (eds) Medical Image Computing and Computer-Assisted Intervention (MICCAI 2015). Lecture Notes in Computer Science, vol 9351. Springer, Cham., 2015.

[16] K. He, X. Zhang, S. Ren, J. Sun, "Deep Residual Learning for Image Recognition", In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770-778.

XIX Conferencia de la Asociación Española para la Inteligencia Artificial

I Taller Ciudades inteligentes e Inteligencia Artificial (SCAI)



Citizen Centric Optimal Electric Vehicle Charging Stations Locations in a Full City: Case of Malaga*

*Note: The full contents of this paper have been published in the volume Lecture Notes in Artificial Intelligence 12882 (LNAI 12882)

Christian Cintrano
University of Malaga
Malaga, Spain
cintrano@lcc.uma.es

Jamal Toutouh
University of Malaga, Malaga, Spain
Massachusetts Institute of Technology, CSAIL, MA, USA
jamal@lcc.uma.es,toutouh@mit.edu

Enrique Alba
University of Malaga
Malaga, Spain
eat@lcc.uma.es

***Abstract*—This article presents the problem of locating electric vehicle (EV) charging stations in a city by defining the `Electric Vehicle Charging Stations Locations` (`EV-CSL`) problem. The idea is to minimize the distance the citizens have to travel to charge their vehicles. `EV-CSL` takes into account the maximum number of charging stations to install and the electric power requirements. Two metaheuristics are applied to address the relying optimization problem: a genetic algorithm (GA) and a variable neighborhood search (VNS). The experimental analysis over a realistic scenario of Malaga city, Spain, shows that the metaheuristics are able to find competitive solutions which dramatically improve the actual installation of the stations in Malaga. GA provided statistically the best results.**

***Index Terms*—Electric vehicle, Charging station location, Metaheuristics**

DOI: https://doi.org/10.1007/978-3-030-85713-4_24

Estudio de la calidad de servicio de autobuses empleando datos abiertos: caso de Melilla

Christian Cintrano
ITIS Software
Malaga, Spain
cintrano@lcc.uma.es

José Ángel Morell
ITIS Software
Malaga, Spain
jamorell@lcc.uma.es

Enrique Alba
ITIS Software
Malaga, Spain
eat@lcc.uma.es

Jamal Toutouh
ITIS Software
Malaga, Spain
jamal@lcc.uma.es

***Resumen*—Analizar de forma cuantitativa y cualitativa el servicio de transporte público es crucial para identificar sus limitaciones y así ofrecer maneras de optimizarlo. En este trabajo se presenta el análisis del servicio de autobuses de Melilla empleando datos abiertos. Estos datos son importantes porque su acceso está disponible para todo el mundo y no requieren la generación de nuevas fuentes de información. La herramienta propuesta en este trabajo puede generalizarse para ser utilizada en cualquier otra ciudad empleando diferentes métodos de optimización para mejorar el servicio de autobuses ofertado. Los resultados del análisis realizado utilizando los indicadores de calidad propuestos muestran que la calidad del servicio de autobuses de Melilla varía según el distrito. Los distritos periféricos reciben una mejor calidad de servicio. Sin embargo, los distritos más pobres, no disponen de una calidad de servicio superior como sería deseable ya que son los que tienen mayores dificultades para acceder al transporte privado.**

***Index Terms*—Red de autobuses, Análisis de la ciudad, Smart cities.**

I. Introducción

El creciente aumento del tráfico rodado privado en las ciudades no solo está generando problemas medioambientales [1], sino que la generación de ruido y gases contaminantes [2] asociados al mismo están teniendo un impacto directo sobre la salud de las personas [3]. Según la Comisión Europea, la contaminación atmosférica es el principal peligro para la salud los ciudadanos en la Unión Europea [4]. Por otro lado, según la Agencia Europea de Medio Ambiente, la exposición a elevados niveles de ruido genera un alto riesgo para la salud, causando unas 16.600 muertes prematuras al año en el continente [5].

Por lo tanto, reducir el tráfico por carretera en las ciudades es una cuestión relevante. Una forma eficaz para conseguirlo es la de mejorar la calidad del servicio del transporte público y en concreto, de los autobuses urbanos [6]. Es necesario disponer de una red de transporte público eficaz y de calidad que permita a los ciudadanos prescindir en lo posible del transporte privado. Sin embargo, el ciudadano suele percibir el servicio de autobuses como poco fiable [7], con tiempos de viaje impredecibles y horarios que no se cumplen provocados en la mayoría de de los casos a un tráfico poco fluido.

Se han publicado diferentes investigaciones en las que se identifican las causas que influyen en la calidad del servicio del sistema público de autobuses urbanos tales como la duración del trayecto, la frecuencia, la comodidad de los pasajeros, el precio y la seguridad de los vehículos [8]–[11]. Estos pueden estar originados por un diseño deficiente o por la congestión de la red producida por una alta demanda. Todos ellos pueden ser ajustados por las autoridades para fomentar el transporte público en detrimento del privado.

Los diversos factores que influyen en un diseño eficiente y de calidad del servicio de transporte público lo convierten en un problema complicado. Así, dentro del marco de las *Smart Cities* han surgido diferentes estudios que aplican Inteligencia Computacional para el diseño de la red de autobuses [12]–[16]. Estos trabajos abordan el diseño de la red como un problema de optimización y proponen soluciones que tienen en cuenta elementos de la infraestructura (por ejemplo, ubicación de las paradas), la flota de vehículos, la frecuencia de los viajes, etc.

Este tipo de trabajos de optimización hacen uso de datos tanto abiertos como cerrados (por ejemplo, de la propia empresa que provee el servicio de autobuses). Los datos abiertos están siendo fundamentales en el desarrollo de las *Smart Cities* [17]. Con ellos podemos evaluar distintos aspectos de la ciudad y de la vida de sus habitantes y diseñar aplicaciones que asistan tanto a los ciudadanos como a los gestores [18]. Así, en el estado del arte encontramos trabajos que aplican datos abiertos a problemas de movilidad y eficiencia en el transporte [19]; salud y polución [2], [20], [21]; y diseño urbano y salud [22].

En este trabajo hemos desarrollado una herramienta para el análisis de la calidad del servicio de autobuses en las ciudades utilizando datos abiertos [23] llamada *Bus Analisys for Optimization System* (`BusAnOS`). Emplea sistemas de información geográfica, estadística y métodos de aprendizaje automático dando como resultado una solución que permite cuantificar múltiples indicadores de la calidad del servicio de autobuses. Posteriormente, se pueden utilizar estos resultados para aplicar métodos inteligentes, como algoritmos de optimización, que provean de forma automática de una red óptima de autobuses.

El objeto de análisis de este trabajo es la ciudad de Melilla, el territorio autónomo más pequeño de España con una superficie de 12 km^2 y una población de 87.076 en el año 2020 [24]. Dispone de 6 líneas de autobuses urbanos, un número que podría parecer suficiente para el tamaño de su territorio. Sin embargo, las líneas ofrecen una escasa eficiencia y sufren problemas de regularidad y accesibilidad. En este trabajo, estudiamos diferentes aspectos del servicio de autobuses de la ciudad como la distribución de las paradas o la ventaja en tiempo que ofrecen respecto a recorrer los trayectos caminando.

También realizamos un análisis sobre la relación entre la calidad del servicio, la densidad de población y la realidad socioeconómica entre los diferentes distritos.

Las principales contribuciones de este trabajo son: *i)* proponer una serie de indicadores para evaluar la calidad del servicio de autobuses empleando únicamente datos abiertos, *ii)* implementar un software que permite el análisis del transporte público y *iii)* analizar el sistema de autobuses de Melilla.

El resto del artículo está organizado de la siguiente manera: la Sección II describe nuestro caso de estudio, la ciudad de Melilla; la Sección III presenta los datos abiertos y los diferentes indicadores utilizados; la Sección IV muestra el análisis empírico realizado del servicio de autobuses de Melilla; finalmente, la Sección V presenta las conclusiones de este estudio y algunos trabajos futuros.

II. Caso de estudio: servicio público de autobuses en Melilla

Esta sección introduce el caso de estudio: la Ciudad Autónoma de Melilla y su servicio público de autobuses.

A. Ciudad Autónoma de Melilla

La ciudad de Melilla es el territorio autónomo más pequeño en extensión de España con una superficie de 12 km^2 y una población de 87.076 en el año 2020 [24], su densidad de población es superior a la de ciudades como Madrid, Sevilla o Málaga [25]. Está situada al norte de Marruecos (África) a orillas del mar Mediterráneo. Existe una gran diversidad cultural en la ciudad debido a su situación geográfica y también a su historia. Desde su incorporación en el siglo XV hasta nuestros días han convivido comunidades con culturas y religiones diferentes como cristianos, musulmanes, judíos e hindúes.

La ciudad limita al este con el mar Alborán, al norte y al oeste con las comunas de Mariguari y Farhana y al sur con la ciudad de Beni-Enzar, estas últimas pertenecientes a Marruecos. La división administrativa está compuesta por 8 distritos, 25 barrios y 35 segmentos censales [25], [26].

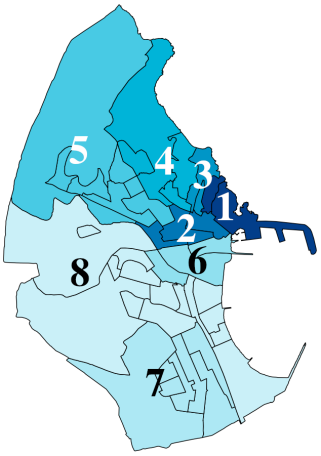


Figura 1. Ubicación de los distritos y segmentos.

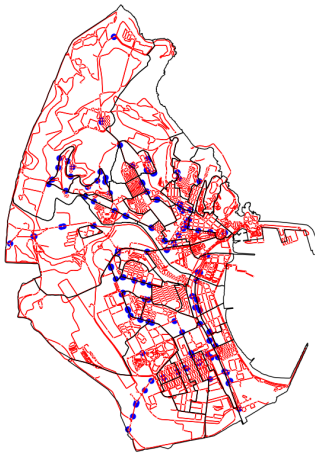


Figura 2. Calles (rojo) y ubicación de las paradas de autobús (azul).

B. Servicio de autobús ofertado

El servicio de autobuses en Melilla lo ofrece la Cooperativa Omnibus de Autobuses (COA). Es la única concesionaria de estos servicios y está compuesta por los antiguos transportistas de viajeros de la ciudad. Existen 6 líneas de autobuses urbanos, articuladas por una flota de 16 vehículos más los de reserva.

La Tabla I resume los datos del servicio principal de autobuses durante los días laborables que es cuando su uso es mayor. Las columnas distancia, paradas y frecuencia contienen la distancia que recorren los autobuses, el número de paradas totales y el tiempo en minutos entre la salida consecutiva de dos autobuses en una misma parada.

Tabla I
Información general de las principales líneas de la COA.

línea	distancia (km)	paradas	frecuencia media	horario de servicio
L1	9,6	37	20/30	7:00-22:00
L2	7,6	21	10/15	7:00-22:00
L3	14,3	43	20/30	7:00-22:00
L5	8,8	20	30	7:00-22:00
L6	7,0	29	40	7:00-22:00
L7	7,0	15	15/20	7:00-21:00

A pesar de ser Melilla un territorio pequeño, las líneas padecen deficiencias de regularidad y accesibilidad, debido principalmente al tráfico privado, generando numerosos retrasos y unos niveles bajos del servicio, con una utilización de un 7 % respecto al 93 % del transporte privado. Los principales problemas a los que debe enfrentarse son [27] *(i)* una inadecuada jerarquía de la red viaria urbana que fuerza a los autobuses a compartir los mismos carriles que los vehículos privados, *(ii)* un inadecuado modelo de estacionamiento de vehículos que aumenta la siniestralidad, *(iii)* y la falta de paradas donde los usuarios puedan estar informados de los recorridos y horarios de las diferentes líneas.

III. Datos abiertos y métricas analizadas

Una vez presentado el caso de estudio vamos a proceder a introducir las diferentes fuentes de datos abiertos empleadas. Además, vamos a describir las métricas utilizadas para medir la calidad de servicio de los autobuses en los diferentes distritos de la ciudad de Melilla.

A. Datos abiertos empleados en el análisis

A la hora de realizar cualquier estudio sobre movilidad inteligente, es necesario disponer de datos realistas sobre la ciudad en cuestión. Hoy en día las ciudades disponen de portales en donde publican multitud de datos abiertos. Además, existen webs e iniciativas en las que la propia comunidad de usuarios es la que se encarga de recoger y publicar datos abiertos. A continuación describiremos nuestro caso de estudio y las fuentes de datos utilizadas.

En este trabajo nos vamos a centrar en el estudio de la ciudad de Melilla, España. En la Tabla II reportamos diferentes datos relativos a cada uno de los ocho distritos de la ciudad (ver Figura 1), su área en km^2, el número de viviendas y densidad de población en cada uno de los distritos en el 2020.

Tabla II
DATOS DE POBLACIÓN DE LOS DISTRITOS DE LA CIUDAD DE MELILLA.

distrito	área	viviendas	población	densidad	pobreza	paradas
1	0,30	770	3117	10390,0	34,4	0
2	0,46	1455	5517	11993,5	48,1	16
3	0,06	676	2744	45733,3	48,1	1
4	0,56	1665	8941	15966,1	50,1	15
5	1,14	3206	15484	13582,5	69,5	28
6	0,58	1113	4782	8244,8	34,3	6
7	1,46	2025	12943	8865,1	29,6	26
8	3,25	7016	30229	9301,2	17,1	36

El número de habitantes fue obtenido del padrón municipal del Instituto Nacional de Estadística [24]. Para nosotros también es importante analizar los aspectos socioeconómicos de la población, por ello, hemos incluido el nivel de pobreza de cada uno de los distritos en este estudio. Con estos datos queremos comparar la calidad del servicio entre los barrios con mayor poder adquisitivo y los más desfavorecidos. Los datos se han extraído de los informes realizados por la Red Europea de Lucha contra la Pobreza y la Exclusión Social en Melilla [28]. Cabe destacar que el distrito 8 es el más grande en cuanto a extensión y población y además es el que tiene un menor índice de pobreza. Por otro lado, el distrito 5 es el segundo más poblado pero tiene el mayor índice. Además, y ya que estamos estudiando la calidad del servicio de la red de autobuses, reportamos el número de paradas en cada distrito.

En el caso de la movilidad, Open Street Maps [29] se ha convertido en una fuente abierta de información cartográfica ampliamente utilizada. En este trabajo, hemos utilizado el mapa de carreteras descargado de Open Street Maps para obtener la información relativa a carriles, vías, sentidos de las calles, etc. Una de las particularidades de Melilla, es que no existen carriles destinados a la circulación del transporte público en exclusiva. Todos los vehículos circulan por las mismas vías. Esto hace que el estado del tráfico (horas punta, atascos etc.) afecte en gran medida a la calidad del servicio de la red de autobuses.

Además del mapa de carreteras, hemos utilizado información proveniente de la empresa de autobuses[1] de Melilla. De la web de la empresa de autobuses hemos conseguido los datos de líneas, paradas, frecuencias y horarios. Con todo esto queremos dar un estudio realista y actual de la ciudad. En la Figura 2 se muestran el mapa de carreteras de la ciudad y la ubicación de las distintas paradas de autobús. Podemos observar que hay áreas de la ciudad, como la zona norte, que tienen una menor concentración de paradas que otras zonas de la ciudad. Es interesante destacar que esa zona corresponde al distrito 5 en el que más de dos tercios de su población es pobre. El transporte público es especialmente importante para las personas con pocos recursos. En la Sección IV-C analizaremos más detalladamente la relación existente entre el índice de pobreza y la red de autobuses de Melilla.

[1]COA Melilla: https://coamelilla.com/

B. *Métricas analizadas*

En este trabajo queremos estudiar la calidad de servicio de la red de autobuses de Melilla. Para poder cuantificar dicha calidad de servicio es necesario definir una serie de métricas. A continuación presentamos las diferentes métricas utilizadas en este estudio:

- Distancias y tiempos de recorrido a pie. Esta métrica es útil para evaluar el beneficio en tiempo que supone emplear el transporte público frente a ir caminando.
- Numero de paradas de autobús en cada segmento censal de cada distrito
- Densidad de paradas por distrito: paradas por km^2
- Distancia caminando desde un segmento censal a la parada de bus mas cercana
- Tiempos de viaje en autobuses (se tiene en cuenta la ruta en autobús y se hacen transbordos si interesa)
- Métrica de calidad de servicio 1: Distancias caminando de un segmento censal de la ciudad a cualquier otro
- Métrica de calidad de servicio 2: Mejora del tiempo de viaje usando el autobús con respecto a ir caminando

Para localizar a la ciudadanía se utilizó la información proveniente de los diferentes segmentos censales. Así, nuestra unidad básica de población es el segmento, siendo un distrito un conjunto de segmentos. Para localizar geográficamente al segmento se utilizó el centroide del mismo. Esto nos permitió obtener la información de los posibles trayectos que pudiera realizar un ciudadano cualquiera en la ciudad.

En el cálculos de las distancias se ha empleado el algoritmo de Dijkstra para obtener el camino más corto, por calle o carretera, para los ciudadanos o los autobuses.

Utilizando estas distancias y las velocidades de los autobuses y ciudadanos calculamos los diferentes tiempos de viaje. Ya que este es un primer estudio de la calidad de servicio no hemos utilizado simuladores de tráfico y hemos considerado la velocidad del autobús constante mientras está en movimiento. Además, hemos estimado el tiempo que gasta el autobús en cada parada para la subida y bajada de pasajeros. Dicho tiempo de espera es de 1 min. Por otro lado, para los peatones se ha supuesto una velocidad de 4,5 km/h de media [30].

Para calcular todas estas métricas hemos utilizados datos abiertos provenientes de las distintas fuentes presentadas previamente en la Sección III-A.

C. *Herramientas para el análisis*

Uno de los aportes de este trabajo es el desarrollo de `BusAnOS`[2], un software desarrollado en `Python` que permite evaluar las métricas presentadas anteriormente de forma automática. Está implementado usando bibliotecas específicas para sistemas de información geográfica (como Geopandas y Arcpy), de análisis de grafos (como OSMnx y NetworX) y de análisis de datos (como Numpy, scikit-learn y Pandas).

IV. ANÁLISIS EMPÍRICO DEL SERVICIO DE AUTOBUSES

Esta sección presenta el estudio de las diferentes métricas presentadas con anterioridad.

[2]`BusAnOS` - https://github.com/jamaltoutouh/melilla-bus-analysis

A. Movilidad peatonal

La Tabla III resume la movilidad a pie entre los distintos segmentos de la ciudad. Muestra las distancias y los tiempos de traslado caminando entre todos los segmentos censales (agrupados por distritos). Los tiempos de desplazamiento se han aproximado utilizando como base que una persona camina a unos 4,5 km/h de media [30]. Como se puede observar, el reducido tamaño de Melilla hace que el mayor tiempo de desplazamiento entre dos segmentos sea de 68,2 minutos. Por otro lado, los distritos céntricos 2 y 6 (ver Figura 1) son aquellos cuyos vecinos tienen una distancia más corta a la hora de desplazarse por la ciudad. Así, los habitantes del distrito 7 son aquellos que se encuentran más alejados, y por lo tanto, necesitan más tiempo para desplazarse.

De acuerdo con esto, cabría esperar que el servicio de autobuses proporcionara soluciones competitivas a los distritos que se encuentran más en la periferia.

Tabla III
DISTANCIAS (METROS) Y TIEMPOS DE VIAJE CAMINANDO (MINUTOS).

distrito	distancia (metros)		tiempo (minutos)	
	media±desv	max	media±desv	max
1	2072,6±849,3	4258,4	27,6±11,3	56,8
2	1589,9±759,1	3680,5	21,2±10,1	49,1
3	1946,3±902,0	4080,1	26,0±12,0	54,4
4	2278,9±1073,5	5114,9	30,4±14,3	68,2
5	2274,5±1080,2	5110,8	30,3±14,4	68,1
6	1606,4±705,1	3721,4	21,4±9,4	49,6
7	2389,8±1171,0	5043,7	31,9±15,6	67,3
8	1951,7±936,2	5114,9	26,0±12,5	68,2

B. Instalación de las paradas

La instalación de las paradas de autobús es crítica a la hora de ofrecer el servicio. Las mismas tienen que estar lo suficientemente distribuidas por los distritos y el número tiene que ser el suficiente para que los usuarios no tengan que caminar demasiado. A su vez, no se pueden instalar demasiadas paradas por el coste de instalación y porque condicionan el tiempo de viaje de los autobuses.

Como se puede observar en la Tabla II, a medida que los distritos aumentan en tamaño, el número de paradas instaladas en los mismos también es mayor. Haciendo un análisis de Pearson se confirma dicha relación con una correlación resultante de 0,86 (i.e., correlación alta). Una relación similar aparece cuando se correlaciona el número de habitantes de los distritos y el de paradas instaladas, obteniendo un coeficiente de Pearson de 0,91 (i.e., correlación alta).

La Tabla IV muestra la densidad de paradas (es decir, el número de paradas por km^2) y la distancia que tienen que recorrer los vecinos de los diferentes segmentos a la parada de autobús más cercana, agrupado por distritos. Como se puede observar, existe una variabilidad importante en la densidad de paradas de autobús instaladas.

El distrito 1 no tiene ninguna parada de autobús instalada (recordemos que es el segundo distrito más pequeño y el de menos población de la ciudad, ver Tabla II). Así, la población del distrito 1 es la que tiene que recorrer una mayor distancia hasta llegar a la parada de autobuses más cercana. Siendo esta de casi 500 metros de media. El distrito 2 es el que tiene una mayor densidad de paradas y la distancia que tienen que recorrer sus vecinos es la menor de todas (131 metros de media).

Tabla IV
DENSIDAD DE PARADAS Y DISTANCIA CAMINANDO (METROS).

distrito	paradas/km^2	distancia a parada más cercana		
		media±desv	min	max
1	0,0	489,7±205,6	284,1	695,3
2	34,8	131,2±49,6	62,5	178,1
3	16,7	197,7±133,2	64,4	330,9
4	26,8	339,5±243,1	132,5	748,2
5	24,6	176,5±125,5	24,2	456,1
6	10,3	231,4±85,8	127,4	337,6
7	17,8	201,3±51,8	121,5	299,8
8	11,1	354,1±466,6	89,5	424,4

Se ha incluido en esta medida de calidad de servicio (la distancia que hay que recorrer hasta la parada más cercana) la población que tiene que hacer dicho camino. Así, se ha multiplicado la distancia a recorrer por el numero de habitantes del segmento y se ha dividido por el número total de población (a forma de normalizar esta métrica de calidad).

Tras una primera evaluación, se ha realizado un estudio de agrupación mediante *K-Means* y el resultado nos refleja que existen tres grandes grupos (según el método del codo para la búsqueda de grupos). El primero incluye al distrito 1; el segundo a los distritos 3, 6, 7 y 8; y el tercero a los distritos 2, 4 y 5 (ver Figura 3). Como se puede observar por la dispersión en el eje-y (distancia ponderada), los tres grupos ofrecen una calidad de servicio similar. Sin embargo, el tercer grupo necesita instalar un mayor número de paradas de autobuses, por lo que la decisión de donde ubicar las paradas dentro de los distritos condiciona la calidad de servicio. Finalmente, se puede comprobar que no existe una correlación entre el número de paradas y la distancia ponderada puesto que Pearson retorna un coeficiente de correlación de -0,17 (i.e., correlación negativa muy débil).

Este resultado nos indica que una selección adecuada de la ubicación de las paradas empleando métodos inteligentes puede ser determinante para proveer un servicio de transporte público de calidad [15], [16].

C. Mejora respecto a ir caminando

En esta sección analizamos la calidad del servicio de los autobuses en términos de la mejora del tiempo de viaje con respecto a ir caminando. De esta forma, se han calculado todos los posibles traslados entre segmentos de la ciudad, teniendo en cuenta que el usuario puede ir a pie, subirse a un autobús, volver a caminar y subirse a otro autobús en otra parada. De este modo, el usuario toma la ruta más rápida (de menor tiempo de viaje) de entre todas las posibilidades.

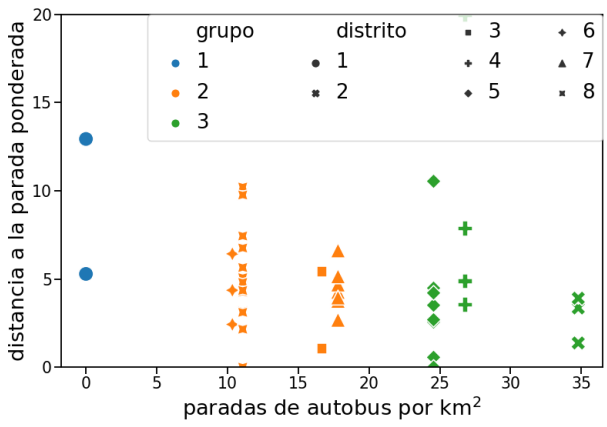


Figura 3. Resultado del estudio de la distancia a la parada más cercana ponderada y la densidad de paradas de autobuses (*K-Means*).

La frecuencia de llegada de los autobuses a las paradas determina el tiempo mínimo y máximo que los usuarios deben esperar, y esto condiciona el tiempo total de viaje. Nosotros realizamos un análisis teniendo esto en cuenta. Marcamos los tiempos de espera como fracción de tiempo de espera respecto al tiempo total entre la llegada de dos autobuses consecutivos. La fracción 0,00 indica que el usuario no tiene que esperar nada y la 1,00 que ha tenido que esperar el tiempo máximo. las fracciones 0,25, 0,50, y 0,75 representan que el usuario espera un 25 %, un 50 % y un 75 % del tiempo máximo entre autobuses, respectivamente. La Tabla V resume los resultados obtenidos mostrando las mejoras por tiempo (en forma de media±desviación típica) agrupadas por distrito y tiempo de espera.

Tabla V
MEJORAS PORCENTUALES SOBRE EL TIEMPO DE VIAJE CAMINANDO EN FUNCIÓN DEL TIEMPO DE ESPERA (MEDIA±DESVIACIÓN ESTÁNDAR).

distrito	fracción de tiempo de espera				
	0,00	0,25	0,50	0,75	1,00
1	46,4±2,4	25,9±1,5	14,0±3,3	8,1±3,8	4,9±2,6
2	54,9±1,0	27,7±2,9	17,1±1,9	12,8±0,9	9,4±0,3
3	57,2±7,2	35,5±5,0	22,3±3,3	15,5±2,3	10,41±2,2
4	51,3±5,9	29,6±5,2	17,2±3,5	11,6±2,6	8,74±1,7
5	**58,7±10,6**	**38,5±13,8**	**25,2±13,2**	**17,7±11,0**	**12,62±8,4**
6	54,0±9,4	31,2±8,7	20,2±6,8	14,0±5,3	10,2±4,0
7	56,4±1,9	36,2±3,4	22,9±4,0	15,4±3,4	10,4±2,5
8	48,9±11,4	25,6±11,0	14,6±9,3	8,8±7,2	5,8±5,3

Los distritos 5 y 7 son los que muestran una mayor mejora en los tiempos de viaje. Esto es importante puesto que, como se puede ver en la Figura 1, son dos distritos periféricos y también los más distantes (ver las distancias en Tabla III).

Si el usuario llega justo cuando está entrando el autobús a la parada (fracción de tiempo de espera de 0,00), las mejoras en los tiempos de viaje media son considerables (son siempre mayores de un 45 %). Tal y como se puede ver en la Tabla V, existe un impacto significativo respecto a tener que esperar o no al autobús. Para casi todos los distritos, la mejora del uso del autobús disminuye en más de un 40 % si el usuario llega justo cuando el autobús se ha marchado (fracción de tiempo de espera de 1,00).

Los autores de este trabajo consideramos que el servicio de transporte público debe dar respuesta a todos los ciudadanos. Sin embargo, creemos que este servicio debe prestar especial atención a aquellos colectivos que tengan menos posibilidades económicas para poder acceder al transporte privado. Es por ello que hemos analizado la mejora en los tiempos de viaje en relación al nivel de pobreza de los distritos (medido en porcentaje de habitantes pobres).

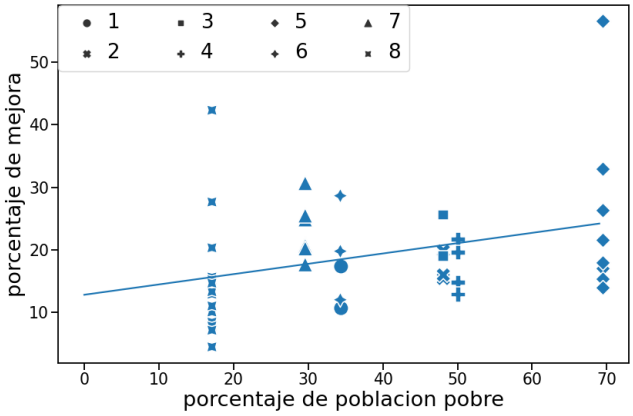


Figura 4. Resultado del estudio de la mejora del tiempo de viaje empleando el autobús y el nivel de pobreza del distrito. La línea representa la recta de regresión que mejor ajusta los datos.

Haciendo el análisis de correlación entre los indicadores del porcentaje de mejora en los tiempos de viaje y el nivel de pobreza obtenemos un coeficiente de correlación de Pearson de 0,33. Esto indica la existencia de una correlación medio baja. Lo deseable sería una mayor correlación.

La Figura 4 muestra el porcentaje de mejora (eje-y) con respecto al porcentaje de habitantes pobres en los distritos. Se observa que el distrito 8, el de menor nivel de pobreza, es el que tiene en general una mejora inferior. No obstante, el distrito 7, el segundo con mayor nivel de pobreza, muestra niveles similares de mejora que los demás distritos.

A su vez, hemos calculado la recta de regresión que confirma que no existe una clara relación entre el nivel de pobreza y la mejora que ofrece el servicio de autobuses (ver Figura 4). Como se puede observar, la recta es prácticamente paralela al eje-x de la gráfica. Lo ideal sería que la propia recta de regresión mostrara una pendiente mayor, significando que los distritos reciben una calidad de servicio de transporte público en función de la necesidad de los ciudadanos.

V. CONCLUSIONES Y TRABAJO FUTURO

Este trabajo está motivado por la necesidad de proporcionar un servicio de transporte público eficiente y de calidad que represente una alternativa al transporte privado. En concreto, analizamos el servicio de autobuses de Melilla. Proponemos

el uso de datos abiertos y describimos la importancia de este análisis para identificar las debilidades y así poder optimizarlo.

Se ha comprobado que Melilla, al ser una ciudad pequeña, las distancias que suelen recorrer los peatones no son mayores de 5,2 km, lo que supone que no caminan más de 70 minutos. Se ha estudiado la localización de las paradas de autobuses, y como métrica de calidad de servicio, la distancia que tiene que recorrer un usuario hasta encontrar la parada más cercana. Se ha comprobado que hay un distrito que no tiene ninguna parada instalada (distrito 1). Esto hace que los vecinos de este distrito tengan que caminar una mayor distancia hasta llegar a la parada de autobús más cercana (una media de 489 metros). A su vez, hemos comprobado que la densidad de paradas instaladas (número de paradas instaladas por km^2) en un distrito no está relacionado con la distancia que recorren los vecinos. Debido a esto, es importante elegir de forma inteligente la ubicación de las paradas para reducir la distancia media que recorren los usuarios.

Analizando la mejora que ofrece el traslado en autobús frente a ir caminando como métrica de calidad de servicio, hemos observado que los distritos 5 y 7 muestran una mayor mejora en los tiempos de viaje. Esto es importante puesto que son dos distritos periféricos. A su vez, se ha comprobado que esta mejora, en general, no es mayor en los distritos con mayores índices de pobreza (lo deseable).

Estos resultados indican que la red de autobuses de Melilla podría ofrecer un servicio más eficiente según los indicadores analizados empleando herramientas inteligentes para su diseño.

Las principales líneas de trabajo futuro son: *i)* aplicar algoritmos de optimización que empleen `BusAnOS` como método de evaluación para proponer mejoras en el servicio de autobuses, *ii)* incluir nuevas variables como el precio, *iii)* extender el análisis realizado incluyendo segmentos de población distintos agrupados por edad y/o nivel socioeconómico, *iv)* incluir en el análisis información pormenorizada de los tiempo de trayecto en función del tráfico y la velocidad media real en las vías.

Referencias

[1] Dirección General de Tráfico (DGT) , " El impacto medioambiental del tráfico ," https://www.dgt.es/PEVI/documentos/catalogo_recursos/didacticos/did_adultas/impacto.pdf, 2014.

[2] I. Lebrusán and J. Toutouh, "Using Smart City Tools to Evaluate the Effectiveness of a Low Emissions Zone in Spain: Madrid Central," *Smart Cities*, vol. 3, no. 2, pp. 456–478, 2020.

[3] N. Soni and N. Soni, "Benefits of pedestrianization and warrants to pedestrianize an area," *Land use policy*, vol. 57, pp. 139–150, 2016.

[4] European Commission, "Directive 2008/50/ec of the european parliament and of the council of 21 may 2008 on ambient air quality and cleaner air for europe," *Official Journal of the European Union, L*, vol. 152, pp. 1–44, 2008.

[5] Carlos Nicolás Fraile , " Ruido: 16.000 muertes al año. El 80 % causado por el tráfico. ," https://revista.dgt.es/es/noticias/internacional/2019/0425Dia-contra-el-ruido.shtml, 2019.

[6] F. González, V. Valdivieso, L. De Grange, and R. Troncoso, "Impact of the dedicated infrastructure on bus service quality: an empirical analysis," *Applied Economics*, vol. 51, no. 55, pp. 5961–5971, 2019.

[7] X. Chen, L. Yu, Y. Zhang, and J. Guo, "Analyzing urban bus service reliability at the stop, route, and network levels," *Transportation research part A: policy and practice*, vol. 43, no. 8, pp. 722–734, 2009.

[8] J. G. Strathman and J. R. Hopper, "Empirical analysis of bus transit on-time performance," *Transportation Research Part A: Policy and Practice*, vol. 27, no. 2, pp. 93–100, 1993.

[9] M. D. Abkowitz and I. Engelstein, "Factors affecting running time on transit routes," *Transportation Research Part A: General*, vol. 17, no. 2, pp. 107–113, 1983.

[10] C. Carrion and D. Levinson, "Value of travel time reliability: A review of current evidence," *Transportation research part A: policy and practice*, vol. 46, no. 4, pp. 720–741, 2012.

[11] E. I. Diab, M. G. Badami, and A. M. El-Geneidy, "Bus transit service reliability and improvement strategies: Integrating the perspectives of passengers and transit agencies in north america," *Transport Reviews*, vol. 35, no. 3, pp. 292–328, 2015.

[12] A. Kumar, P. Srikanth, A. Nayyar, G. Sharma, R. Krishnamurthi, and M. Alazab, "A novel simulated-annealing based electric bus system design, simulation, and analysis for dehradun smart city," *IEEE Access*, vol. 8, pp. 89 395–89 424, 2020.

[13] T. Nguyen, D. Q. Nguyen-Phuoc, and Y. Wong, "Developing artificial neural networks to estimate real-time onboard bus ride comfort," *Neural Computing and Applications*, vol. 33, no. 10, pp. 5287–5299, 2021.

[14] E. Köksal Ahmed, Z. Li, B. Veeravalli, and S. Ren, "Reinforcement learning-enabled genetic algorithm for school bus scheduling," *Journal of Intelligent Transportation Systems*, pp. 1–19, 2020.

[15] E. Fabbiani, S. Nesmachnow, J. Toutouh, A. Tchernykh, A. Avetisyan, and G. Radchenko, "Analysis of mobility patterns for public transportation and bus stops relocation," *Programming and Computer Software*, vol. 44, no. 6, pp. 508–525, 2018.

[16] D. G. Rossit, S. Nesmachnow, and J. Toutouh, "Multiobjective design of sustainable public transportation systems," in *1st International Workshop on Advanced Information and Computation Technologies and Systems*. CEUR-WS, 2021, pp. 152–159.

[17] S. Barns, "Smart cities and urban data platforms: Designing interfaces for smart governance," *City, culture and society*, vol. 12, pp. 5–12, 2018.

[18] E. Estrada, R. Maciel, A. P. P. Negrón, G. L. López, V. Larios, and A. Ochoa, "Framework for the analysis of smart cities models," in *International Conference on Software Process Improvement*. Springer, 2018, pp. 261–269.

[19] S. Šurdonja, T. Giuffrè, and A. Deluka-Tibljaš, "Smart mobility solutions–necessary precondition for a well-functioning smart city," *Transportation research procedia*, vol. 45, pp. 604–611, 2020.

[20] I. Lebrusán and J. Toutouh, "Car restriction policies for better urban health: a low emission zone in madrid, spain," *Air Quality, Atmosphere & Health*, vol. 14, pp. 333–342, 2020.

[21] J. Toutouh, I. Lebrusán, and S. Nesmachnow, "Computational Intelligence for Evaluating the Air Quality in the Center of Madrid, Spain," in *International Conference on Optimization and Learning*. Springer, 2020, pp. 115–127.

[22] I. Lebrusán and J. Toutouh, "Smart city tools to evaluate age-healthy environments," in *Smart Cities*, S. Nesmachnow and L. Hernández Callejo, Eds. Cham: Springer International Publishing, 2021, pp. 285–301.

[23] F. R. Soriano, J. J. Samper-Zapater, J. J. Martinez-Dura, R. V. Cirilo-Gimeno, and J. M. Plume, "Smart mobility trends: Open data and other tools," *IEEE Intelligent Transportation Systems Magazine*, vol. 10, no. 2, pp. 6–16, 2018.

[24] Instituto Nacional de Estadística (INE) , " Melilla: Población por municipios y sexo ," https://www.ine.es/jaxiT3/Datos.htm?t=2909, 2020.

[25] Consejería de medio ambiente , " Mapa Estratégico de Ruido de la Ciudad Autónoma de Melilla ," https://www.melilla.es/melillaportal/RecursosWeb/DOCUMENTOS/1/0_11640_1.pdf, 2014.

[26] Consejería de Economía, Empleo y Turismo , " Las condiciones de vida de la población pobre de la Ciudad de Melilla. Un estudio comparativo 1995-2009 ," https://eapnmelilla.files.wordpress.com/2011/01/estudio-sobre-la-pobreza-en-la-ciudad-de-melilla.pdf, 2009.

[27] Conserjería de Medio Ambiente y Sostenibilidad. Ciudad autónoma de Melilla. , " El sistema de transporte público de Melilla. ," https://medioambientemelilla.es/el-sistema-de-transporte-publico/, 2020.

[28] EAPN Melilla, "Red Europea de Lucha contra la Pobreza y la Exclusión Social en Melilla," https://eapnmelilla.wordpress.com/, 2021, Accessed: 2021-01-05.

[29] OpenStreetMap contributors, "Planet dump retrieved from https://planet.osm.org ," https://www.openstreetmap.org, 2017.

[30] R. L. Knoblauch, M. T. Pietrucha, and M. Nitzburg, "Field studies of pedestrian walking speed and start-up time," *Transportation Research Record*, vol. 1538, no. 1, pp. 27–38, 1996.

A Collaborative Platform for the Detection of Non-inclusive Situations in Smart Cities*

*Note: The full contents of this paper have been published in the volume Lecture Notes in Artificial Intelligence 12882 (LNAI 12882)

Alfonso González-Briones, Iván García-Magariño, Jorge J. Gómez-Sanz, Rubén Fuentes-Fernández, Juan Pavón
Research Group on Agent-Based, Social and Interdisciplinary Applications (GRASIA)
Complutense University of Madrid
Madrid, Spain
alfonsogb@ucm.es

***Abstract*—The quality of daily life is a key factor in today's urban development. This is a complex aspect, as different groups of citizens understand it quite differently and have different needs. This is particularly important for people with disabilities. When thinking about solutions for smart cities, there are a few approaches for the identification of the problems or obstacles that their implementation poses for different groups of people. Addressing this issue requires tools that provide an interdisciplinary perspective, so the proposed smart solutions really promote the improvement of the quality and well-being of all citizens in cities. In this line, this paper presents a tool that facilitates collaboration among stakeholders through identifying and analysing the elements that prevent a smart city solution from being inclusive. The platform is based on visual novels that illustrate how the solution would work in real settings. Users comment on the novel, play with it to discover and evaluate the solution, and propose changes. Then, a natural language processing engine analyses their comments, and returns the identified non-inclusive situations, ideas for improvement or solutions, and the degree of affinity that each participant has with each of them. The tool has been validated in a mobility scenario. There, three groups of people with different needs used the tool to identify obstacles and non-inclusive situations. The results have been used to assess and improve the tool.**

***Index Terms*—Inclusive Smart cities, Reduced mobility, Visual novels, Natural Language Processing, Collaborative Design, Functional diversity.**

DOI: https://doi.org/10.1007/978-3-030-85713-4_20

Análisis del tráfico de búsqueda de redes WiFi de los Smartphones y sus prestaciones para la monitorización de personas en ciudades inteligentes.

1st A. Fernández-Ares
Departamento de Teoría de la Señal, Telemática y Telecomunicaciones
ETSIIT - Universidad de Granada
Granada, Spain
antares@ugr.es

2nd A. Mora-García
Departamento de Teoría de la Señal, Telemática y Telecomunicaciones
ETSIIT - Universidad de Granada
Granada, Spain

3rd M. García-Arenas
Departamento de Arquitectura y Tecnología de Computadores
ETSIIT - Universidad de Granada
Granada, Spain

4th P. García-Sánchez
Departamento de Arquitectura y Tecnología de Computadores
ETSIIT - Universidad de Granada
Granada, Spain

5th P.A. Castillo-Valdivieso
Departamento de Arquitectura y Tecnología de Computadores
ETSIIT - Universidad de Granada
Granada, Spain

***Resumen*—Los dispositivos inteligentes como los smartphones se han convertido en una gran fuente de datos a usar en las ciudades inteligentes. En este artículo se presenta un análisis de las tramas de red WiFi emitidas de forma inadvertida por estos dispositivos para su utilización para la monitorización y posicionamiento.**

***Index Terms*—dispositivos inteligentes, smartphone, WiFi, monitorización, redes inalámbricas**

I. Introducción

La situación de pandemia vivida actualmente ha puesto de manifiesto la necesidad de las ciudades inteligentes de monitorizar la localización de sus ciudadanos con el fin de limitar aforos o trazar las personas con las que han estado en contacto. Los smartphones han supuesto un excelente aliado, debido a que el ciudadano promedio lo considera un objeto prácticamente imprescindible para sus salidas de su domicilio.

Estos dispositivos se consideran inteligentes por su capacidad para comunicarse con otros dispositivos de igual o distinta naturaleza, haciendo uso de redes inalámbricas para ello, entre ellas WiFi. El uso de estas redes genera un tráfico de red que es fácilmente capturable, procesable y empleable para trazar el movimiento los smartphones a lo largo de la ciudad gracias a nodos sensores que capturen ese tráfico y lo procesen.

En este artículo se presenta un estudio de las comunicaciones WiFi emitidas de forma inadvertida por los smartphones en su búsqueda de redes a las que conectarse. Gracias a este tráfico de red, se puede trackear los desplazamientos y estancias de sus propietarios. Esta información ha sido extraída del sistema Mobywit [1] tras monitorizar más de 13 millones de dispositivos a lo largo de varios años.

A continuación se presentará la metodología de trabajo, con una breve descripción del rendimiento del sistema empleado. Se presentarán experimentos que evalúan el comportamiento de los smartphones en su búsqueda de redes WiFi. Por último, se presentarán las conclusiones extraídas de este trabajo.

II. Metodología

Las comunicaciones inalámbricas WiFi se sustentan en la existencia de un área de cobertura denominada BSS (Basic Service Set) que ofrece un conjunto básico de servicios para las telecomunicaciones. Cuando un dispositivo es vinculado dentro de un BSS puede comunicarse con otros dispositivos pertenecientes a esa BSS. Dependiendo de si existe o no un punto de acceso (access point o AP) en la BSS se habla de BSS de infraestructura o BSS independiente. Los smarpthones hacen uso general de las redes desplegadas por una BSS de infraestructura, siendo el uso de BSS independientes relegado a la configuración de periféricos como cámaras de acción.

Para incorporarse a un BSS, los dispositivos han de localizar la existencia de estas áreas y sus AP asociados. A este proceso se le denomina escaneo de las redes. En función del comportamiento del AP y los dispositivos, se habla de dos métodos de escaneo distintos:

- Escaneo pasivo: en el que los AP envían tramas denominadas beacons anunciando el SSID y otros parámetros radios de la BSS necesarios para la conexión.

Este trabajo ha sido financiado en parte por los proyectos RTI2018-102002-A-I00 (Ministerio de Ciencia, Innovación y Universidades), TIN2017-85727-C4-2-P y PID2020-115570GB-C22 (Ministerio de Economía y Competitividad), B-TIC-402-UGR18 (FEDER y Junta de Andalucía), y P18-RT-4830 (Junta de Andalucía).

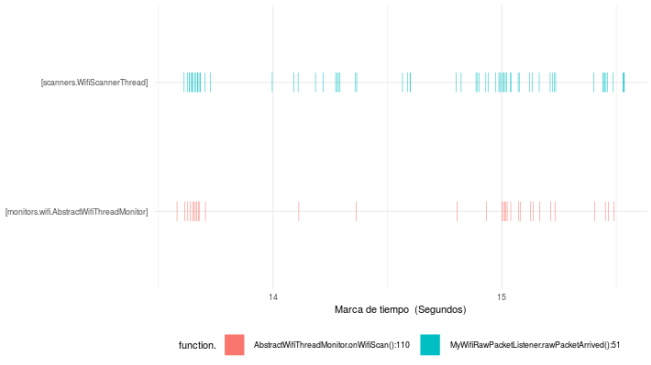


Figura 1. Escaneo de tramas de búsqueda de BSS.

- Escaneo activo: en el que los dispositivos envían una trama denominada Probe Request a una red concreta con un SSID determinado a la que se haya vinculado anteriormente. El AP responde con un mensaje Probe Response en caso de existir correspondencia.

Este tipo de comunicaciones de escaneo activo nos permite determinar que un dispositivo en las inmediaciones está intentando encontrar a las BSS con las que anteriormente se ha vinculado.

Para capturar y procesar estas comunicaciones se ha empleado el sistema Mobywit, para así poder estudiar cada cuanto tiempo emite un smartphone tramas de búsqueda por escaneo activo de forma totalmente inadvertida para su portador. Esta emisión se puede emplear para emplazar espacio-temporalmente a los dispositivos en las inmediaciones del nodo sensor que ha capturado dicha comunicación.

Un nodo Mobywit consta de varios elementos tanto hardware como software, a destacar el escáner y el monitor. El escáner es un proceso vinculado a la tarjeta de red encargado de capturar las tramas y filtrarlas para quedarse únicamente con tramas relacionadas con la búsqueda de BSS, descartando las tramas que sean relacionadas a tráfico de red o de infraestructura. Las tramas de búsqueda de BSS son enviadas al monitor para que actualice la información de monitorización del dispositivo, incluyendo la dirección MAC del dispositivo realizando la búsqueda activa de BSS, para que el AP sea capaz de responder de forma identificable a cada dispositivos.

En la Figura 1 se representa el instante de captura de las tramas emitidas por un smartphone en un entorno controlado y como al monitor solo le llegan aquellas que pertenecen a la búsqueda de BSS.

El tiempo de procesamiento de cada trama es inferior[1] a $1ms$ y el tiempo de envío de la información de la trama del sensor al monitor se mueve entre los 2 y $3ms$. La tarjeta de red empleada tiene una tasa de transferencia de $150Mbps$ ($18{,}75MBps$) según especificaciones del fabricante y el bus USB al que se conecta al nodo dispone de un ancho de banda de $480mbps$ ($60MBps$). El tipo de tramas capturadas tienen tamaños que oscilan entre los $50Bytes$ al máximo teórico de $2304Bytes$, pero se sitúan en torno a los $65bytes$ en la práctica (Tabla I).

Cuadro I
TAMAÑO PROMEDIO DE LAS TRAMAS DE RED CAPTURADAS.

Frame Type	Average Size
Acknowledgemet	$50Bytes$
Probe Request	$150Bytes$
Probe Response	$300Bytes$
Beacon	$300Bytes$
Request-to-send	$50Bytes$

En entornos prácticos, el número de tramas procesadas por segundo oscila entre las $200pfs$ (frames per second) y las $800fps$. Esto supone un tráfico capturado entre los $\sim 12KBps$ y $\sim 50KBps$, muy por debajo de la capacidad de la tarjeta de red.

En laboratorio, empleando simuladores de tráfico[2] se ha logrado tráfico constante de $1\,300fps$ ($\sim 85KBps$) con picos puntuales de hasta $3\,000fps$ ($\sim 200KBps$) soportados por el sistema de monitorización.

En entornos reales muy densamente poblados de dispositivos, el sistema ha procesado unos picos máximos de $\sim 17{,}9KBps$ generados por cientos de dispositivos simultáneos.

El sistema empleado para las mediciones, excede por tanto las capacidades de captura y procesamiento de tramas requeridas para la monitorización, por lo que no es susceptible de no capturar ningún tipo de tráfico de red emitido por el dispositivo por saturación del mismo.

III. EXPERIMENTOS

Una vez presentada la herramienta empleada para la monitorización, se realizan varios experimentos para estudiar el comportamiento de los smartphones en cuanto a la búsqueda activa de BSS, con el fin de poder determinar la precisión de esta fuente de datos.

Se van a realizar cuatro experimentos. El primero de ellos consiste en el estudio de las tramas emitidas por los smartphones en distintos escenarios de uso. El segundo, presenta el efecto de rebote de las tramas WiFi, que implica que una misma trama de red emitida por el smartphone va a ser recibida varias veces por el escáner. El tercero, es una aproximación al uso de la RSSI de la señal WiFi como aproximación a la distancia. Y por último, se debate el uso de MAC de búsquedas aleatorias para la monitorización y su posible impacto en el uso de los smartphones y sus emisiones WiFi como fuente fiable para la monitorización.

III-A. Frecuencia de la búsqueda activa de BSS

Una vez determinado que el sistema de monitorización propuesto es capaz de capturar un flujo constante de tráfico de red, es necesario comprobar cada cuánto tiempo es detectable un dispositivo que emplea WiFi y se encuentra en la búsqueda de redes. La Figura 2 muestra 1 minuto del tráfico de red capturado en las inmediaciones de un nodo de monitorización en pruebas.

[1]Unidad mínima de medida del sistema empleado.

[2]Ostinato - https://github.com/pstavirs/ostinato.

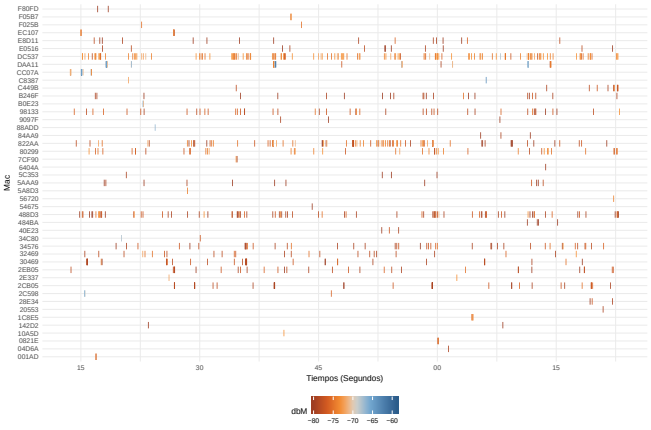


Figura 2. Instantánea del tráfico de red capturado por un nodo de monitorización durante 1 minuto. Se ha limitado el tráfico capturado al recibido por debajo de $-80dBm$.

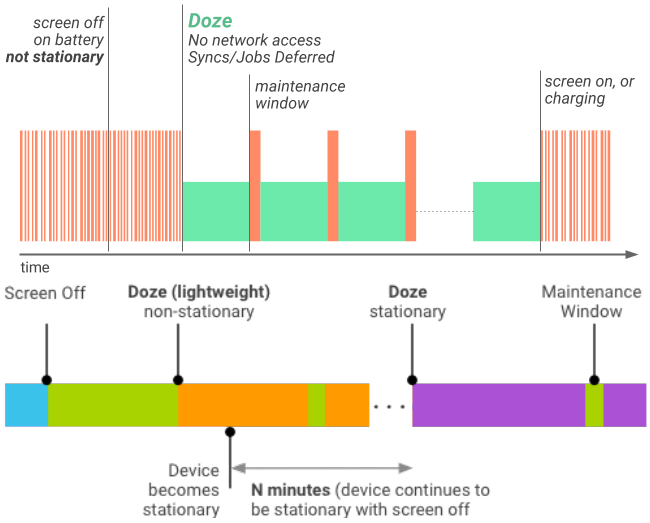


Figura 3. Esquema de funcionamiento del sistema de ahorro de energía `DOZE` empleado en dispositivos ANDROID. El dispositivo mantiene varios modos basado en el movimiento del dispositivo y el estado de la pantalla para apagar las comunicaciones del dispositivo, manteniendo ventanas espaciadas en el tiempo para habilitar nuevamente las comunicaciones.
Fuente: https://source.android.com/devices/tech/power/platform_mgmt

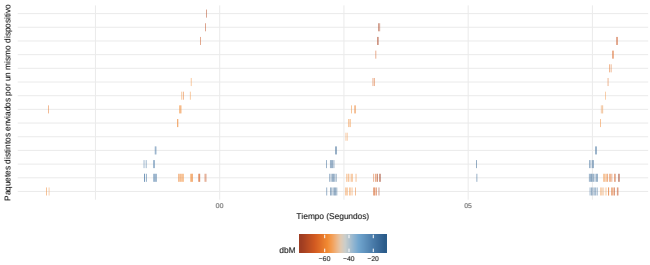


Figura 4. Tráfico WiFi capturado de un dispositivo en movimiento. El dispositivo no se encuentra conectado a ninguna red WiFi. Cada línea horizontal representa un tipo de paquete capturado, agrupando los que sean del mismo tipo y naturaleza en la misma línea horizontal aunque hayan sido emitidos en instantes distintos de tiempo.

Los dispositivos que se detectan de forma reiterativa realizan sus comunicaciones en tiempos inferiores a los 10 segundos. Muchos otros dispositivos son detectados únicamente en un instante de tiempo, no volviendo en esa ventana. Se puede teorizar que o bien son dispositivos en movimiento y ya no se encuentran las inmediaciones del nodo o que han espaciado la búsqueda de redes.

Dado que la búsqueda activa de BSS implica emisión de tráfico de red, la frecuencia de búsqueda varía en función de las opciones de energía del dispositivo. Los dispositivos inteligentes disponen de modos de ahorro de energía que emplean sus sensores para determinar cuándo el dispositivo está en reposo o cuándo se encuentra en movimiento. La mayoría de fabricantes implementa estos mecanismos para aumentar la vida útil de la batería de sus dispositivos, siendo el mejor documentado el sistema `DOZE` empleado en los dispositivos . En la Figura 3 se presenta el funcionamiento del sistema `DOZE`.

Mientras los dispositivos se encuentran en movimiento (`non-stationary`), el dispositivo tiene libre acceso a las comunicaciones. En al Figura 4 se presenta el tráfico de red capturado emitido por un único dispositivo a lo largo de 10 segundos. Este dispositivo de ejemplo se encuentra con la pantalla apagada, no conectado a ninguna red WiFi y en el bolsillo de su portador. Aproximadamente cada 2 segundos se detectan tramas de red emitidas por el dispositivo inteligente para la búsqueda de un BSS al que vincularse. Además, este proceso se realiza incluso aunque el dispositivo se encuentre ya conectado a una BSS, pues cambiará de BSS si esta le ofrece una mejor cobertura.

En el momento en que el dispositivo queda en absoluto reposo (`stationary`), como por ejemplo al posicionarlo encima de una superficie horizontal, entran en funcionamiento los sistemas de ahorro de energía. Al estar el dispositivo con la pantalla apagada y en reposo, el tiempo habilitado para las comunicaciones incluida la búsqueda de redes se espacia de forma incremental, hasta superar los 10 segundos, como se recoge en la Figura 5, en lo que se conoce como ventana de mantenimiento o `Maintenance Windows` en el sistema `Doze`. El sistema de ahorro de energía asume que si el usuario no está en movimiento, no encontrará nuevas BSS que le den cobertura por lo que espaciará la búsqueda en el tiempo.

Una de las peculiaridades del sistema de búsqueda de BSSS de los dispositivos inteligentes, es que se realiza incluso aunque el dispositivo se encuentre con el WiFi apagado o en el modo avión. En ese caso, las peticiones se espacian mucho más en el tiempo, del orden de 5 minutos con la pantalla apagada y en reposo, pero se activan automáticamente al encender la pantalla, mover el dispositivo o realizar cualquier desplazamiento o cualquier otra acción que requiera una interacción con el dispositivo o cambie al estado `non-stationary`. Esto es debido a que los principales desarrolladores de sistemas operativos para dispositivos móviles, les interesa este comportamiento, pues la detección de redes WiFi es empleada

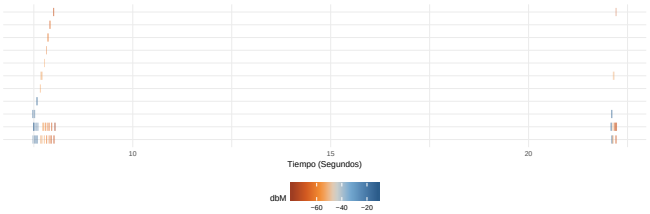


Figura 5. Tráfico WiFi capturado de un dispositivo en absoluto reposo, sobre una superficie horizontal estática. Entran en funcionamiento los modos de ahorro de batería y se espacian las comunicaciones y búsqueda de redes.

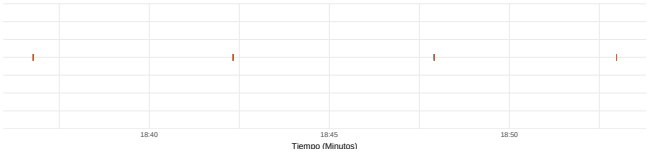


Figura 6. Tráfico WiFi capturado de un dispositivo en reposo y con el modo avión activado.

por los dispositivos para mejorar su posicionamiento. La Figura 6 representa las tramas capturadas por un dispositivo en reposo en modo avión, con una frecuencia de búsqueda situada entre los 5 y 6 minutos.

Para evitar esto, los dispositivos móviles inteligentes suelen tener un opción o varias muy escondidas para deshabilitar la búsqueda de redes WiFi:

- En los dispositivos Android, se tiene que activar el modo de localización por solo GPS, deshabilitar la búsqueda de redes WiFi en cualquier momento para mejorar la ubicación, deshabilitar la interfaz WiFi y deshabilitar el modo `Usar Wifi en suspensión`; encontrándose cada opción en un menú distinto.
- En IOS, se tienen que deshabilitar los servicios de red WiFi en los ajustes de privacidad relativos al servicio de localización, deshabilitar la interfaz WiFi, deshabilitar la notificación de nuevas redes y deshabilitar las opciones de llamadas WiFi de la aplicación de teléfono.

Estas opciones resultan demasiado específicas como para ser activadas sin intencionalidad por el grueso de usuarios. Además, con ella solo se impide que sea el sistema operativo el que realice la búsqueda de redes WiFi. Multitud de aplicaciones solicitan el permiso de gestión de redes inalámbricas obteniendo permiso por defecto por parte de la inmensa mayoría de los usuarios. Solicitan la gestión de las redes inalámbricas pues emplean esta información de forma periódica para usos internos, basados principalmente en la detección o no de ciertas redes. De esta forma, aplicaciones en apariencia tan simples como las pertenecientes a Starbucks, Burguer King o cualquier otra tienda que ofrezca de conexión WiFi a sus clientes estará configurada para detectar redes WiFi en la proximidad, pues es el mecanismo que emplean para determinar que el dispositivo se encuentra en uno de sus comercios, el ofrecimiento de una red WiFi con un determinado `SSID` común a estos establecimientos.

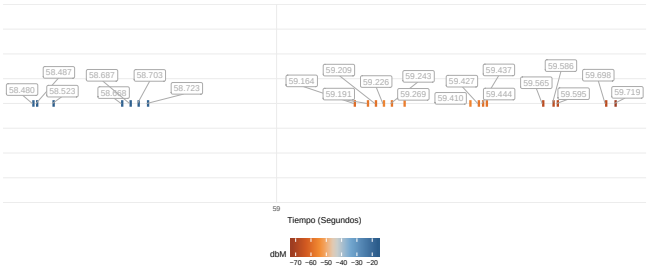


Figura 7. Detalle del efecto de reflexión de las tramas capturadas por WiFi. Una misma trama de red emitida es detectada hasta 20 veces por el nodo de monitorización.

De esta forma, la única manera efectiva de que un dispositivo inteligente como un smartphone no sea detectado por medio de la captación de tramas de red WiFi de búsqueda activa de BSS, es que se encuentre apagado. En cualquier otro escenario, requerirá una gestión plena y dedicada de la configuración del dispositivo, lejos del alcance de la mayoría de usuarios.

III-B. Efecto de la reflexión de WiFi en la monitorización.

Debido a que se capturan tramas propagadas por el aire por medio del protocolo WiFi y a la reflexión de las ondas en el medio, una misma trama de búsqueda activa de BSS emitida por un dispositivo es susceptible de ser capturada en varias ocasiones por el nodo de monitorización. Es por ello que en las figuras del estudio anterior se presentaban varias detecciones del mismo tipo de trama en corto espacio de tiempo y distintas intensidades. Este fenómeno se puede observar en más detalle en la Figura 7, donde se observa cómo una única trama emitida ha sido detectada en más de 20 ocasiones en un periodo de algo más de 1 segundo en un entorno aislado y controlado de laboratorio.

La naturaleza de reflexiones en las comunicaciones WiFi escapa del ámbito de estudio de este artículo, sin embargo el uso de smartphones para la monitorización debe ser tolerante con este fenómeno físico ya que detectar varias veces la misma trama supone un incremento de la carga de trabajo del monitor, que tendrá que actualizar en multitud de ocasiones el instante de última detección de ese dispositivo, una por cada vez que la trama en cuestión es detectada de forma adicional. Además, este fenómeno puede producir imprecisiones en el sistema.

Es deseable que el monitor sea capaz de determinar si la información que le facilita el sensor pertenece a una reflexión de una trama del dispositivo ya detectada anteriormente o es una trama enviada nuevamente por el dispositivo. Se observa en la Figura 8 la cantidad de veces que la misma trama WiFi emitida por un dispositivo inteligente en reposo (emitiendo una trama cada 10 segundos aproximadamente) es detectada en un nodo ubicado en un entorno controlado.

En promedio las tramas permanecen detectables en el medio hasta 1 segundo de tiempo después de la primera recepción, por lo que una de las posibilidades del monitor sería reducir la sensibilidad al segundo, para no considerar tramas recibidas en

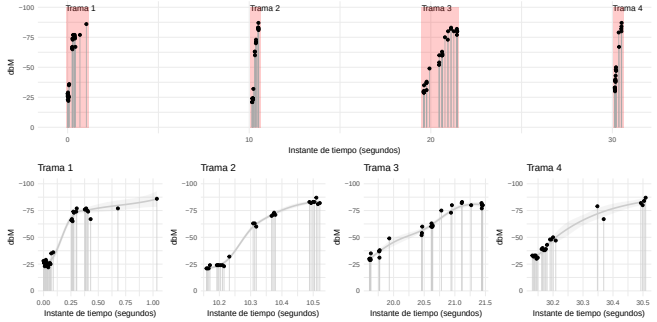


Figura 8. Efecto de reflexión de las tramas capturadas. Para facilitar la interpretación, la gráfica de arriba muestra la información en la línea continua de tiempo, y las gráficas inferiores los valores precisos de cada trama.

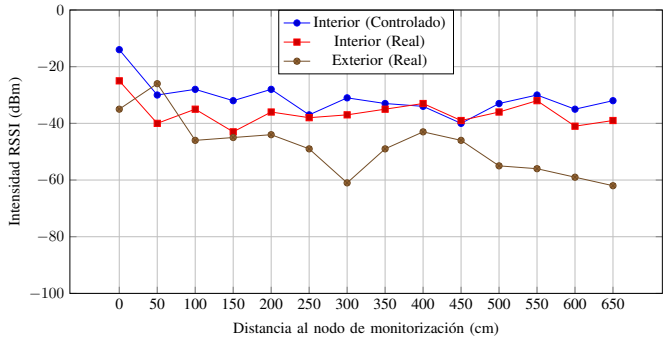


Figura 9. Relación entre la distancia al nodo de monitorización y la intensidad RSSI.

instantes de tiempo inferiores al segundo como reincidencias de un dispositivo. Además la intensidad RSSI de las recepciones sucesivas de una misma trama empeora respecto a la primera que se ha recibido.

Por lo tanto, es sencillo discernir empleando ambos métodos cuando una trama capturada es una nueva trama para el sistema o se trata de una reflexión de una ya procesada, sin tener que volver a procesarlas o realizar algún análisis de su contenido.

III-C. Uso del RSSI para la estimación de distancia

Si bien la relación entre la intensidad `RSSI` y la distancia está bien estudiada y modelizada [2]–[5] normalmente se basa en la intensidad con la que un dispositivo móvil detecta los distintos puntos de acceso en una zona acotada geográficamente. En este caso, se plantea el escenario contrario siendo un nodo el que detecta un único dispositivo emitiendo tramas de red WiFi, por lo que no es posible la triangulación ni ningún mecanismo de posicionamiento preciso. La Figura 9 recoge la intensidad `RSSI` de detección mínima de una única trama emitida por un dispositivo inteligente en relación a la distancia al nodo de monitorización en distintos escenarios.

En entornos reales y distancias cortas, existen numerosos factores que influyen más que la distancia en la intensidad de recepción, como son los obstáculos, la posición del dispositivo respecto al portador, la cantidad de dispositivos emitiendo

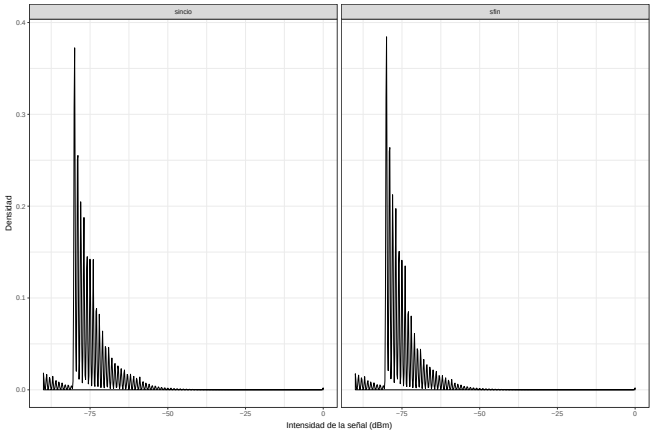


Figura 10. Densidad poblacional de las intensidades de captación de las tramas que determinan la ventana temporal de los dispositivos WiFi detectados por el sistema.

simultáneamente, la presencia de puntos de acceso WiFi o las condiciones climáticas.

En la Figura 10 se presentan los histogramas de las intensidades de primera y última detección de todos los pasos de dispositivos detectados por el sistema.

La mayoría de los pasos detectados se enmarcan entre los $-80dBm$ y $-50dBm$ lo que ofrece una distancia aproximada al nodo entre los 6 y 12 metros.

III-D. Uso de MAC Aleatorias para el escaneo activo

En los últimos `smartphones` de Apple y Android [3] se incorporan mecanismos por el que la dirección MAC empleada en las búsquedas activa de BSS no es la MAC real dispositivo, si no una MAC generada pseudo-aleatoriamente.

Esto supone en principio un duro impacto negativo para el uso de la emisión de tramas WiFi de dispositivos inteligentes como fuente de datos, pues origina dos potenciales problemas. El primero, es que rompe la trazabilidad del dispositivo. El segundo, es que el número de dispositivos detectados se puede ver alterado si en cada búsqueda el dispositivo en corto tiempo emplea una MAC distinta.

Sobre la trazabilidad, según se ha observado hasta la fecha, la dirección MAC aleatoria se establece en el proceso de arranque del sistema operativo (tanto IOS comoo Android) y no se cambia hasta que el dispositivo sufre un `hard reset` [6]–[8]. De esta manera, aunque la dirección MAC empleada en la búsqueda no sea la real, es siempre la misma hasta que el dispositivo no se reinicie completamente.

Este tipo de reinicio habitualmente es el que se realiza cuando el dispositivo se queda sin batería por un largo periodo de tiempo. En dispositivos Android, incluso se ha documentado que ante un reinicio sin batería la MAC asignada vuelve a ser la misma ya que se almacena en el sistema tras solicitarla a un servidor que la establece.

[3] https://www.androidpolice.com/2019/04/05/android-q-randomizes-mac-addresses-by-default-with-per-network-customization/

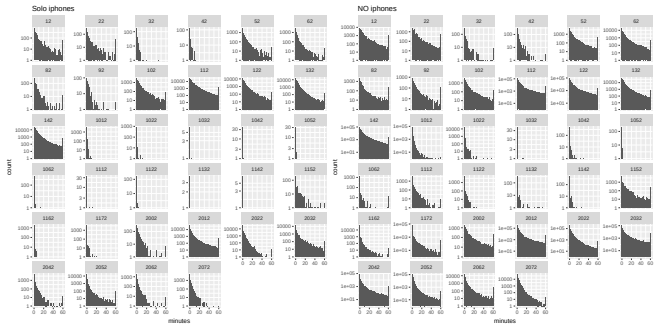


Figura 11. Impacto de la búsqueda con dirección MAC aleatoria en la duración de la estancia.

De igual manera, como el dispositivo no cambia la dirección MAC (empleada en la búsqueda) hasta que se reinicia, empleará siempre la misma en búsquedas sucesivas, por lo que no será detectado con distintas direcciones MAC en cortos periodos de tiempo.

Aunque estos mecanismos de aleatorización buscan no permitir la trazabilidad del dispositivo, su aplicación más directa es no vincular dicha trazabilidad al tráfico de red de un determinado dispositivo. Así aunque un dispositivo con una MAC aleatoria se haya desplazado y haya sido detectado por varios puntos de acceso, cuando el dispositivo se conecte a uno de estos, hará uso de su dirección real. De esta manera, no se podrá vincular la dirección MAC a la que se le ha otorgado la dirección IP (y por tanto el tráfico de red asociado a esta) con la trazabilidad del desplazamiento del dispositivo.

Así pues, la aleatorización de las direcciones MAC no supone un contratiempo al sistema de monitorización propuesto tal y como lo implementan los fabricantes en la actualidad. La generación de una dirección MAC válida del rango del fabricante que no esté en uso por otro dispositivo no es una acción trivial computacionalmente, por lo que no puede ser empleada para aleatorizar la dirección MAC de cada búsqueda.

En la Figura 11 se presentan las duraciones de las estancias de detección de los distintos nodos del sistema para dispositivos considerados iPhone con MAC aleatorias y para el resto de dispositivos.

La evidencia de la gráfica es también evaluada con el test Kolmogorov-Smirnov [9] ($D^+ = 0{,}48267$ y $p-value < 2{,}2\times 10^{-16}$) para determinar si ambas muestras siguen la misma distribución de probabilidad.

IV. Conclusiones

Aunque WiFi no es un protocolo orientando a la detección de dispositivos en la inmediaciones, es posible emplear estas comunicaciones emitidas de forma inadvertida por los dispositivos inteligentes como los smartphones para monitorizar y trazar los desplazamientos de estos dispositivos.

El sistema de monitorización empleado es capaz de procesar en laboratorio flujos constantes de tráfico de red de hasta 3000 tramas por segundo o $200KBps$. En entornos reales el flujo de red que se ha detectado es inferior a los $20KBps$, por lo que la eficiencia máxima del sensor está por encima de lo esperable en entornos reales.

Aunque los dispositivos inteligentes tienen sistemas de ahorro de batería que les hacen interrumpir las comunicaciones cuando el dispositivo se encuentra en reposo, para dispositivos en movimiento el tiempo de búsqueda es inferior a los 5 segundos en todos los casos estudiados. Al detectar el dispositivo que se encuentra en reposo, se espacia esta búsqueda hasta los 10 segundos, incrementándose progresivamente a lo largo del tiempo de inactividad. Incluso desactivando la red WiFi y activando el modo avión, un dispositivo en movimiento o en reposo continuará buscando puntos de acceso cercanos y por tanto emitiendo tramas de búsqueda activa de BSS.

Si bien la reflexión de las tramas es un factor a tener en cuenta puede ser determinados tanto por el tiempo de recepción como por la intensidad RSSI. Esta intensidad `RSSI` puede dar una métrica aproximada de la distancia a la que se encuentra el dispositivo detectado del nodo sensor.

Por tanto, no existen impedimentos para el uso como sistema de monitorización útil para las ciudades inteligente el empleo de la detección de los dispositivos inteligentes por medio de la captación de sus comunicaciones WiFi emitidas de forma inadvertida en la búsqueda activa de BSS.

Referencias

[1] A. Fernández-Ares, A. Mora, M. Arenas, P. García-Sanchez, G. Romero, V. Rivas, P. Castillo, and J. Merelo, "Studying real traffic and mobility scenarios for a smart city using a new monitoring and tracking system," *Future Generation Computer Systems*, vol. 76, pp. 163–179, 2017.

[2] A. Bose and C. H. Foh, "A practical path loss model for indoor wifi positioning enhancement," in *2007 6th International Conference on Information, Communications & Signal Processing*, pp. 1–5, IEEE, 2007.

[3] X. Zhu and Y. Feng, "Rssi-based algorithm for indoor localization," *Communications and Network*, vol. 5, no. 02, p. 37, 2013.

[4] J.-Y. Lee, C.-H. Yoon, H. Park, and J. So, "Analysis of location estimation algorithms for wifi fingerprint-based indoor localization," in *Proc. 2nd Int. Conf. Softw. Technol*, vol. 19, pp. 89–92, 2013.

[5] E. Mok and G. Retscher, "Location determination using wifi fingerprinting versus wifi trilateration," *Journal of Location Based Services*, vol. 1, no. 2, pp. 145–159, 2007.

[6] J. Martin, T. Mayberry, C. Donahue, L. Foppe, L. Brown, C. Riggins, E. C. Rye, and D. Brown, "A study of mac address randomization in mobile devices and when it fails," *Proceedings on Privacy Enhancing Technologies*, vol. 2017, no. 4, pp. 365–383, 2017.

[7] M. Vanhoef, C. Matte, M. Cunche, L. S. Cardoso, and F. Piessens, "Why mac address randomization is not enough: An analysis of wi-fi network discovery mechanisms," in *Proceedings of the 11th ACM on Asia Conference on Computer and Communications Security*, pp. 413–424, ACM, 2016.

[8] J. Freudiger, "How talkative is your mobile device?: an experimental study of wi-fi probe requests," in *Proceedings of the 8th ACM Conference on Security & Privacy in Wireless and Mobile Networks*, p. 8, ACM, 2015.

[9] F. J. Massey Jr, "The kolmogorov-smirnov test for goodness of fit," *Journal of the American statistical Association*, vol. 46, no. 253, pp. 68–78, 1951.

XIX Conferencia de la Asociación Española para la Inteligencia Artificial

Competición de Apps basadas en Inteligencia Artificial



Competición CAEPIA-App: Recomendación de Restaurantes Basada en Contenido Visual

Pablo Pérez-Núñez
Centro de Inteligencia Artificial
Universidad de Oviedo
Gijón, España
pabloperez@uniovi.es

Cristina Cuesta Llavona
Escuela Politécnica de Ingeniería
Universidad de Oviedo
Gijón, España
UO239054@uniovi.es

Jorge Díez Peláez
Centro de Inteligencia Artificial
Universidad de Oviedo
Gijón, España
jdiez@uniovi.es

***Resumen*—En este documento se presenta una aplicación cuyo cometido es recomendar restaurantes que sirvan un tipo de plato semejante al que el usuario de la misma indica mediante una imagen. Esto es útil cuando nos desplazamos a otra ciudad y necesitamos recomendaciones de establecimientos en los que degustar algún plato que nos gusta especialmente. Cargando una imagen de dicho plato en la aplicación, obtendríamos una lista de restaurantes recomendados en la zona de forma casi inmediata. Como en la mayoría de los casos no tendremos fotos de nuestros platos preferidos, o simplemente queremos dejarnos aconsejar, la aplicación también nos mostrará una serie de imágenes con platos sugeridos, donde se destacan principalmente aquellos típicos en la gastronomía de la ubicación actual, que pueden resultar desconocidos para el usuario a la vez que apetecibles. Para realizar todo este proceso, se crearán varias redes neuronales convolucionales que se utilizarán para determinar si hay o no comida en una foto, para extraer las características principales del plato de la imagen y para recomendar, a continuación, una lista de restaurantes.**

***Index Terms*—Android, Sistemas de Recomendación, Redes Neuronales Convolucionales, Tensorflow Lite**

I. Descripción de la Aplicación

La aplicación móvil propuesta para esta competición tiene como objetivo principal ofrecer recomendaciones de restaurantes a usuarios que llegan a una nueva ciudad bien sea por turismo, trabajo u otras causas.

Tradicionalmente, las aplicaciones existentes en este campo recomiendan restaurantes a partir de una búsqueda textual o en función de sitios similares que hayamos visitado con anterioridad. En nuestro caso el usuario no posee un historial ni tiene por que conocer los nombres de restaurantes o platos tradicionales del lugar, por tanto, esta aproximación no sería válida. Para solucionar los problemas descritos, nuestra aplicación tendrá como elemento novedoso la forma de obtener esas recomendaciones, puesto que el punto de partida será una única imagen proporcionada por el usuario a modo de término de búsqueda. Con esta funcionalidad se pretende emular una petición del usuario del tipo *"Quiero comer algo como lo que se ve en la imagen"*. En la Figura 1, se puede ver el flujo de funcionamiento de la aplicación.

Este trabajo ha sido financiado por el proyecto PID2019-109238GB-C21 del Ministerio de Ciencia e Innovación de España. La colaboración de Pablo Pérez-Núñez ha sido financiada por el Gobierno del Principado de Asturias mediante el programa de becas predoctorales *Severo Ochoa* (ref. BP19-012)

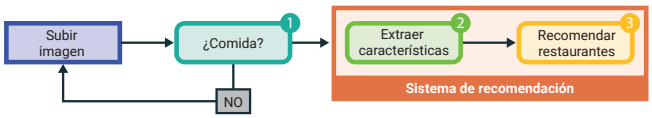


Figura 1. Proceso de recomendación de restaurantes desde que se introduce una imagen hasta que se obtiene la salida.

No solo se le permitirá al usuario utilizar sus propias imágenes, sino que también se le sugerirán, en la pantalla principal de la aplicación, diferentes imágenes de ejemplo. Estas imágenes estarán agrupadas por diversos criterios (comida local, comida italiana, comida rápida,...) con objeto de facilitar la recomendación para casos donde el usuario no disponga de imágenes, no conozca los nombres de los platos o simplemente a modo de inspiración. Es importante destacar el criterio *comida local*, ya que incluye una serie de imágenes de platos que tendrían que variar en función de la localización geográfica del usuario, mostrándole siempre platos típicos del lugar, algo muy útil si se desconoce la gastronomía local.

También se proporcionará acceso desde la propia aplicación a información publicada en Internet de los restaurantes recomendados. Así, el usuario podrá rápidamente ver fotografías tomadas por otros clientes del restaurante además de leer sus críticas. La aplicación recomendará al usuario una lista de restaurantes basándose en el contenido de la imagen (pizza, sushi, pasta,...). Para ello, será necesario saber cuáles son los platos más típicos de cada establecimiento y una vez los conozcamos, recomendaremos solamente aquellos restaurantes especializados en el plato o contenido de la imagen dada por el usuario siguiendo un procedimiento como el representado en la Figura 2.

La idea del desarrollo de esta aplicación surge tras advertir que, muchas de las apps móviles que utilizamos hoy en día, disponen de la capacidad de búsqueda por imagen, pero no se conocía la existencia de ninguna que lo hiciese en el ámbito de la restauración.

Algunos ejemplos de estas aplicaciones pueden ser Google Lens [1] Aliexpress [2] o Amazon [3] que, utilizando esta tecnología, permiten encontrar resultados web o bienes de consumo utilizando solamente la imagen proporcionada por el usuario.

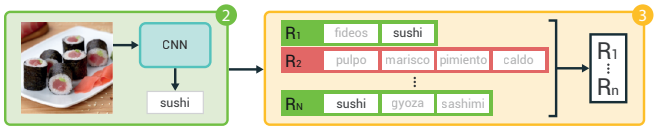


Figura 2. Detalle del procedimiento utilizado para realizar la recomendación. Primero se recibe la imagen del usuario, a continuación se obtienen el contenido de la misma mediante una red convolucional y finalmente recomendamos aquellos restaurantes que ofrezcan el mismo tipo de contenido. Destacar que estos bloques se corresponden con los dos últimos de la Figura 1

II. Técnicas de Inteligencia Artificial empleadas

Con el fin de lograr las funcionalidades descritas anteriormente, es necesario utilizar las siguientes técnicas de IA:

- Redes convolucionales (CNN):
  1. Para clasificar la imagen en comida / no comida.
  2. Para la extracción del plato de la imagen de entrada.
- Sistemas de recomendación
  3. Para generar, a partir de la imagen, una lista de restaurantes recomendados.

La utilidad de estas técnicas durante el proceso de recomendación se puede ver en la Figura 1, correspondiéndose cada uno de los puntos anteriores con los bloques de la misma. Se ha decidido crear un modelo diferente para cada una de las tres funcionalidades que se desean conseguir. A continuación se detallará cada uno de ellos.

II-A. Modelo 1: Clasificar imagen en comida/no comida

Inicialmente, nuestra aplicación ha de verificar que el usuario ha introducido una imagen con comida, esto es necesario para evitar recomendaciones erróneas o confusas por parte del sistema. Se ha de crear, por tanto, un modelo capaz de procesar imágenes a la entrada (utilizando CNNs) y que a la salida resuelva un problema de clasificación binaria, es decir, prediga si hay o no comida en la imagen.

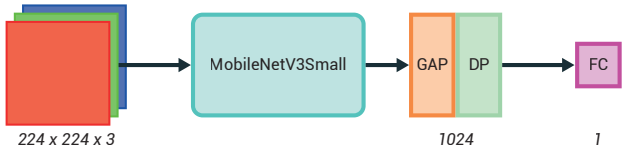


Figura 3. Arquitectura del primer modelo. Entran imágenes de 224×224 en color y se procesan utilizando la parte convolucional de la red MobileNetV3Small. Su salida se transforma mediante una serie de capas que nos permiten reducir la dimensión, controlar el sobreajuste y obtener la predicción final respectivamente. *GAP* representa una capa Global Average Pooling, *DP* Drop Out y *FC* Fully Connected. En la parte inferior de la imagen se puede ver el tamaño de cada capa.

Dado el gran número de capas e hiperparámetros existentes en el campo de las CNN, se ha optado por utilizar una red convolucional existente a la que se aplicará transfer-learning [4] y fine-tuning para evitar crear una arquitectura desde cero. Esto implica aprovecharse de una arquitectura existente que posee buenos resultados resolviendo problemas similares y ajustarla, posteriormente, para nuestro problema concreto.

Teniendo en cuenta que nuestros modelos han de ejecutarse en un dispositivo móvil de la forma más rápida posible, se ha optado por utilizar una red existente con pocas capas y parámetros pensada para ser ejecutada en dispositivos móviles. La red en cuestión es *MobileNetV3Small* [5] y de ella, solamente nos quedaremos con la parte convolucional (extracción de características) a la que añadiremos una serie de capas con el objetivo de resolver el problema de clasificación binaria. El modelo resultante se puede ver en la Figura 3.

Una vez tenemos definida la arquitectura del modelo, necesitamos un conjunto de datos que nos permita entrenarlo y que deberá estar formado por pares del tipo:

$$(imagen,\ ¿comida?). \tag{1}$$

En este caso se va a hacer uso de un conjunto de datos etiquetado manualmente para una investigación previa donde se conoce, para cada una de las más de 18000 imágenes que lo forman, el restaurante donde fueron tomadas y si estas son o no de comida. Son imágenes descargadas de TripAdvisor [6] y pertenecen a todos los restaurantes de la ciudad de Gijón. El conjunto se ha dividido en entrenamiento, validación y test con el fin de poder realizar diversas pruebas de hiper-parámetros y, finalmente, con la mejor combinación se ha entrenado y exportado para su ejecución en la aplicación móvil.

Cabe destacar que este modelo solo se utilizará con aquellas imágenes subidas por el usuario a través de su galería, las imágenes sugeridas por la aplicación no requieren de este control, dado que se han seleccionado previamente asegurándose de que todas contengan comida.

II-B. Modelo 2: Extraer características de la imagen

Tras asegurarnos de que todas las imágenes con las que trabajamos son de comida, podemos continuar con nuestro objetivo: recomendar restaurantes en función del tipo de comida que contiene una imagen dada.

Para lograrlo, no solo necesitamos conocer el tipo de comida que contiene la imagen que nos proporciona el usuario, sino que también necesitamos conocer en qué establecimientos se cocina ese estilo de comida (ver Figura 2). Esta información nos permitirá, en función del tipo de comida que aparezca en la imagen de entrada, seleccionar los restaurantes más adecuados para recomendar al usuario.

La forma más sencilla de obtener el plato de comida presente en una imagen, es crear y entrenar un modelo que nos resuelva esta tarea. Para crearlo, seguiremos la misma filosofía que en el caso anterior, aprovechar una CNN existente modificando la parte final para adaptarla a nuestro problema. Como se puede ver en la Figura 4, la única diferencia con el modelo anterior reside en la capa de salida, donde ahora pretendemos resolver un problema de multi-clasificación en vez de la clasificación binaria previa.

Para entrenar este modelo se ha creado un conjunto con cerca de 100 clases diferentes de comida, conteniendo cada una de ellas aproximadamente 300 imágenes. Inicialmente se definió una lista con todos los tipos de comida que se querían aprender, teniendo en cuenta, no solo la diversidad

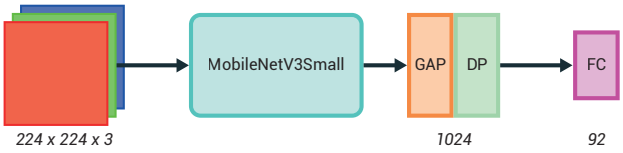


Figura 4. Arquitectura del segundo modelo.

gastronómica de España, sino también la de otros países (Italia, México, ...). Finalmente se implementó un programa capaz de descargar imágenes de Google para cada tipo de comida de la lista, lo que permitió crear un conjunto de forma sencilla. Al igual que en el caso anterior, los datos fueron divididos en entrenamiento, validación y test y, tras varias pruebas de hiperparámetros, el modelo fue entrenado con la mejor combinación conocida.

A partir de este punto ya somos capaces de saber el tipo de comida de una imagen (paso 2 en Figura 2), pero aún necesitamos conocer la misma información para cada uno de los restaurantes de la ciudad (paso 3). Utilizando el modelo anterior sobre el conjunto utilizado en la Sección II-A podemos extraer este conocimiento utilizando el Algoritmo 1.

Algoritmo 1: Tipos de comida de cada restaurante

```
Resultado: {(restaurante, clases de comida populares)}
para todo restaurante r hacer
    Eliminar imágenes de r que no sean de comida;
    para todo imagen i de r hacer
        Predecir con el modelo la clase de comida en i;
    fin
    Quedarse con las clases que más aparecen en r;
fin
```

Una vez conocidas las clases de cada restaurante, ya podemos hacer recomendaciones siguiendo el proceso reflejado en la Figura 2. Teniendo en cuenta que nuestra aplicación se va a ejecutar en un dispositivo móvil, no es lo ideal realizar tanta carga de trabajo en el mismo, por tanto se opta por crear un tercer modelo encargado de, a partir de una imagen retornar directamente la lista de restaurantes más probables, evitándonos así todo el procedimiento intermedio.

II-C. Modelo 3: Generar recomendación de restaurantes

Aprovechando la información extraída en el paso anterior, podemos crear un conjunto de datos que servirá para entrenar el sistema final. Este nuevo conjunto se creará a parir del dataset utilizado en el primer modelo, es decir, el que estaba formado por pares como los indicados en la Ecuación (1).

Como nuestro objetivo es conocer, para cada una de las imágenes del conjunto (que sean de comida), una lista de restaurantes donde se sirva el plato de la imagen, es necesario transformar este conjunto mediante el Algoritmo 2.

Finalmente necesitaremos de un modelo capaz de aprender estas relaciones entre imágenes y restaurantes, lo que solventaremos siguiendo la misma filosofía que en los casos anteriores.

Algoritmo 2: Dataset para el modelo final

```
Resultado: {(imagen, lista de restaurantes)}
Eliminar imágenes que no sean de comida;
para todo imagen i hacer
    Predecir el plato con el Modelo 2.;
    Obtener restaurantes donde sirven dicho plato;
fin
```

Hay que tener en cuenta que en este caso intentaremos resolver un problema multi-etiqueta, puesto que cada imagen de la entrada podrá estar asociada con más de un restaurante a la salida del modelo, en concreto con todos aquellos donde se cocine el plato de la imagen.

Una vez entrenado el modelo, ya podemos exportarlo al dispositivo móvil donde, tras ejecutarlo para una imagen, nos retornará una lista con los restaurantes más probables.

III. Aplicación Móvil

En la Figura 5 podemos ver las secciones esenciales de la aplicación. La captura de la parte izquierda nos muestra la pantalla principal, donde podemos ver diferentes sugerencias (comida local, asiática,...) a modo de inspiración. La idea es que, en función de la ubicación donde se encuentre el usuario, las sugerencias de comida local se adecúen a la zona geográfica concreta. El resto de sugerencias no requieren de modificación alguna y podrían permanecer estáticas para el resto de lugares.

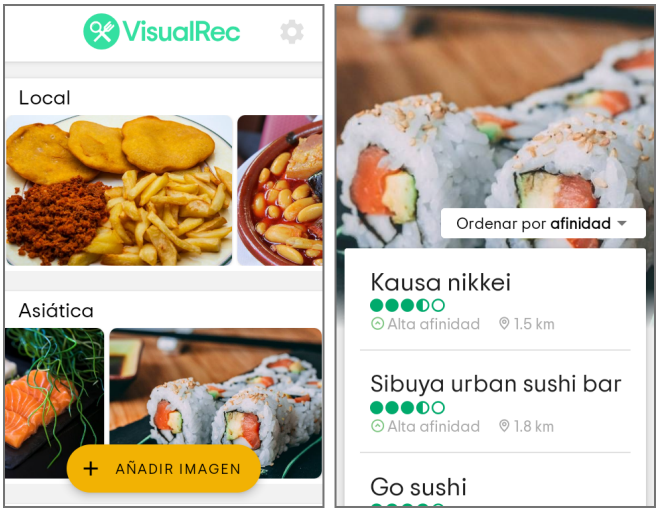


Figura 5. Capturas de la aplicación móvil.

Cuando pulsamos en una de las imágenes sugeridas, la aplicación ejecuta el Modelo 3 sobre la imagen obteniendo una lista de restaurantes como los que se pueden ver en la captura de la parte derecha de la Figura 5. Además de ver las recomendaciones, también podemos ordenarlas por afinidad (probabilidad que nos retorna el modelo), estrellas, nombre o proximidad. Si pulsamos sobre cualquiera de los restaurantes recomendados, la aplicación nos abrirá la web

del mismo dentro de TripAdvisor, donde podremos consultar más información sobre el establecimiento.

Si decidimos subir una foto de nuestra galería en vez de elegir una imagen de entre las sugeridas, la aplicación, antes de mostrarnos directamente la lista de recomendaciones, aplica el Modelo 1 sobre la imagen a fin de verificar que esta contiene comida. En caso negativo se nos muestra un mensaje de aviso y, en caso afirmativo, se procede a la recomendación siguiendo los mismos pasos descritos previamente.

Hemos dejado disponibles para su descarga tanto la aplicación final como el vídeo explicativo[1].

IV. Tecnologías y Herramientas Utilizadas

La aplicación se ha realizado para el sistema operativo Android (dada la accesibilidad de sus herramientas), por ello se utilizará el software de desarrollo Android Studio [7].

Para entrenar, tanto las redes convolucionales como el sistema de recomendación, se ha utilizado la librería Keras [8] sobre TensorFlow ([9], [10]) en el lenguaje de programación Python ([11], [12]). Una vez entrenados los modelos y, para poder realizar la inferencia de los mismos en un dispositivo móvil, se transformaron al formato adecuado para poder ejecutarlos utilizando la librería TensorFlow Lite [13] de Android aplicándoles además las optimizaciones por defecto que permiten simplificarlo y reducir su tamaño.

V. Utilidad , Viabilidad y Escalabilidad

La mayoría de las veces, cuando visitamos una ciudad que no conocemos, no sabemos el nombre exacto del restaurante al que queremos ir, simplemente llegamos a una ciudad y queremos comer algo concreto. En esos casos, esta aplicación nos facilitaría mucho la vida y, por tanto, podríamos decir que tiene una gran utilidad. Prueba de ello es la existencia de aplicaciones con objetivos finales similares como *Instafood* [14] donde se recomiendan restaurantes a partir de un texto y una serie de filtros (precio, distancia,...) pero no a partir de una imagen.

En cuanto a la viabilidad, la aplicación propuesta parece totalmente viable al estar basada en conceptos suficientemente probados a lo largo de los años (redes convolucionales y sistemas de recomendación).

Finalmente se podría decir que estamos ante una aplicación fácilmente escalable. En este artículo se ha centrado todo el trabajo en una ciudad concreta pero, en el caso de que se quisiesen añadir nuevas ciudades, simplemente habría que entrenar un nuevo Modelo 3 (el sistema de recomendación) para dicha ciudad, permitiendo que la aplicación pueda seleccionar el modelo adecuado en función de la ubicación del usuario. Los modelos 1 y 2 podrían aprovecharse para diferentes ciudades sin necesidad de ser reentrenados.

[1]https://onedrive.live.com/?cid=cf10c34991401cd7&id=CF10C34991401CD7%21126679&authkey=!AJ1FRzRKDJRxGDE

VI. Conclusiones y trabajo futuro

En este artículo se ha presentado una aplicación capaz de realizar recomendaciones de restaurantes partiendo simplemente de una foto de comida, la cual puede haber sido proporcionada por el usuario o sugerida por la propia aplicación. Para lograr nuestro objetivo, no nos hemos limitado a encadenar modelos existentes ya entrenados para obtener la solución, si no que se han creado y entrenado modelos desde cero con el fin de resolver nuestros problemas concretos. En total se han creado y entrenado tres redes diferentes, las cuales nos indican, dada una imagen, si tiene comida, el tipo de plato y finalmente los restaurantes donde se sirve dicha comida. Para entrenar cada uno de los modelos se ha tenido que crear un conjunto de datos adecuado a cada problema. Además, se ha ideado un algoritmo de recomendación que, basándose en el contenido de la imagen, es capaz de buscar los restaurantes que más relación tienen con el plato de la misma.

Como trabajo futuro, la primera mejora que se plantea es reducir el número de modelos uniendo el 1 y 2 en un solo sistema, de esta forma tendríamos un único modelo capaz de indicarnos si la foto tiene o no comida y, en caso afirmativo, decirnos de que se trata. Otra mejora sería la automatización de las sugerencias de comida local. Actualmente estas sugerencias se han creado de forma manual para la ciudad concreta con la que se trabaja, pero se cree que se podrían obtener mediante algún procedimiento que las calcule en función de la localización. Finalmente, con el objetivo de añadir cierta transparencia, se podrían indicar los platos destacados de cada restaurante en la lista de recomendaciones, así, el usuario podría verificar la exactitud de las recomendaciones del sistema.

Referencias

[1] "Google lens." http://lens.google. Accedido: 15-06-2021.

[2] A. Mobile, "Aliexpress." http://play.google.com/store/apps/details?id=com.alibaba.aliexpresshd. Accedido: 15-06-2021.

[3] A. M. LLC, "Amazon compras." http://play.google.com/store/apps/details?id=com.amazon.mShop.android.shopping. Accedido: 15-06-2021.

[4] L. Yang, S. Hanneke, and J. Carbonell, "A theory of transfer learning with applications to active learning," *Machine learning*, vol. 90, no. 2, pp. 161–189, 2013.

[5] A. Howard, M. Sandler, G. Chu, L.-C. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan, *et al.*, "Searching for mobilenetv3," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1314–1324, 2019.

[6] "Tripadvisor." www.tripadvisor.es. Accedido: 15-06-2021.

[7] Google and Open Handset Alliance, "Android studio." http://developer.android.com/studio. Accedido: 28-04-2021.

[8] F. Chollet *et al.*, "Keras." keras.io, 2015.

[9] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D. G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, and X. Zheng, "Tensorflow: A system for large-scale machine learning," in *Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation, OSDI 2016*, 2016.

[10] "Tensorflow." www.tensorflow.org. Accedido: 28-04-2021.

[11] G. Van Rossum and F. L. Drake, *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace, 2009.

[12] "Python." www.python.org. Accedido: 28-04-2021.

[13] "Tensorflow lite." www.tensorflow.org/lite. Accedido: 28-04-2021.

[14] V. D. A. D. Company, "Instafood." http://play.google.com/store/apps/details?id=com.randomappsinc.instafood. Accedido: 28-04-2021.

CAEPIA-APP Competition: GimmeHop

Ignacio Huitzil
Artificial Intelligence Research Institute (IIIA-CSIC)
Bellaterra, Spain
ihuitzil@iiia.csic.es

Fernando Alegre
Entertainment Solutions
Zaragoza, Spain
fernandoalegrem@gmail.com

Fernando Bobillo
I3A, University of Zaragoza
Zaragoza, Spain
fbobillo@unizar.es

***Abstract*—This paper overviews GimmeHop, a beer recommendation system for Android mobile devices which uses several Artificial Intelligence techniques, including fuzzy logic, ontologies, and machine learning.**

***Index Terms*—recommender systems, ontologies, fuzzy logic, clustering**

a) Motivation: Beer market is a hot topic which is receiving a notable attention in the last years: more foreign beers are imported by many stores and bars, the number of artisan beers is growing significantly, and users are willing to try new beer styles, thanks in part to the phenomenon of home-brewing. Thus, tools providing users with good recommendations about beers seem very interesting.

b) Description: GimmeHop is a beer recommendation system for Android mobile devices [1]. It is able to deal with user context (in particular, user location), and user preferences. It supports three types of queries or searches:

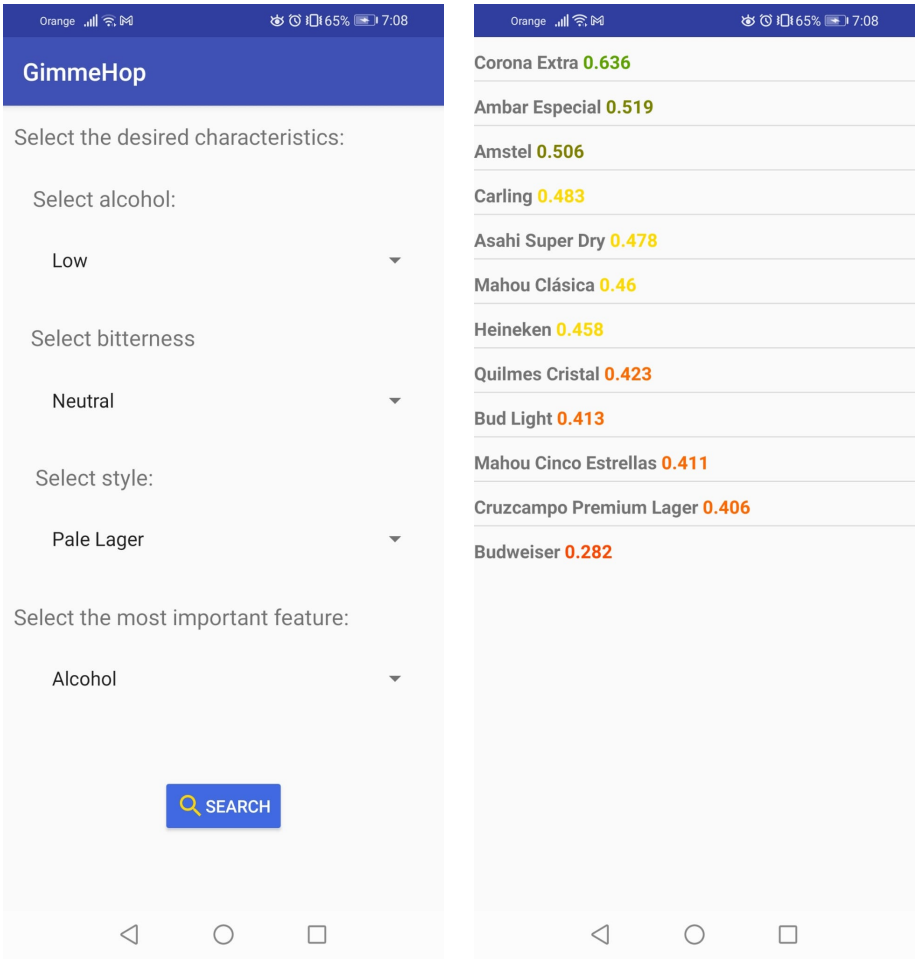
- *Basic search*. The input is a name of a beer or a brewery, and the output is a list of beers or breweries matching syntactically.
- *Advanced search*. The input is a beer type and, optionally, values of some features, namely ABV (alcohol level) and/or IBU (bitterness), and the output is an ordered list of beers together with the recommendation degrees (i.e., the satisfiability degree of the query). Degrees are colored to illustrate the importance of the recommendation (going from green to red as the quality of the recommendation degree decreases). Figure 1 shows an example of advanced search query and the answer of the system.
  The values are expressed using linguistic labels (e.g., *Low* alcohol). Moreover, the user is able to select some user preferences (for example, which is the most important property for his/her).
  Note that the system also takes into account the rating of each beer, but this is not shown in the user interface because users are interested in items with the best possible rating (except perhaps if they host a dinner with their mother-in-law).
- *Similarity search*. The input is a beer, and the output is an ordered list of similar beers together with their similarity degrees.

We were partially supported by the projects TIN2016-78011-C4-3-R (AEI/FEDER, UE), PID2020-113903RB-I00 (AEI/FEDER, UE), and DGA/FEDER.

When the user clicks on any beer, another page displaying basic information about the beer (name, brewery, style, bitterness, alcohol, country, and rating) is presented, as shown in Figure 2 (a).

The main features of GimmeHop are the following ones:

- *It uses a knowledge base (Fuzzy Beer ontology) with* 15317 *individuals* (beers and breweries). The app version submitted to the competition is restricted to 30 beers, as the license of the beer data, owned by a third-party beer company, does not currently make it possible to distribute the data. The schema is very rich to support future extensions of the tool to take into account other features of the beers in the recommendation (for example, color or foam). The fuzzy ontology is encoded using Fuzzy OWL 2 language [2].
- *It supports reasoning.* We use classical ontology reasoners to discover implicit knowledge and then perform some fuzzy reasoning to solve the flexible queries. The app is integrated with HermiT v1.3.8 and TrOWL v1.5.L reasoners. HermiT is an adapted version for Android OS, previously ported by members of our research group [3]. TrOWL reasoner can be directly imported in Android projects, but some tasks are solved not supported by TrOWL and are solved using the OWL API (for example, retrieving the values of the data properties of each individual is supported by Hermit but not by TrOWL).
- *It can solve the queries in local or remote modes*, using a local semantic reasoner or an external one, respectively. A local reasoner is usually slower, but it does not require a good Internet connection and is more appropriate for sensible data.
- *The user interface uses linguistic labels* rather than numbers for some properties (ABV and IBU). This helps the end user to choose the preferences in each feature. For example, in a hot day users typically prefer to ask for a *Lager* style beer with *Low* alcohol. According to our ontology, *Corona Extra* beer, with ABU = 4.5, is associated to the linguistic label *LowAlcohol* with degree 0.59.
- *It uses user context (location).* The app obtains the location (country) to give a higher score to local beers. It gets the location via network or GPS. using Android's Network Provider, which is faster and cheaper (in battery consumption) than GPS Provider.



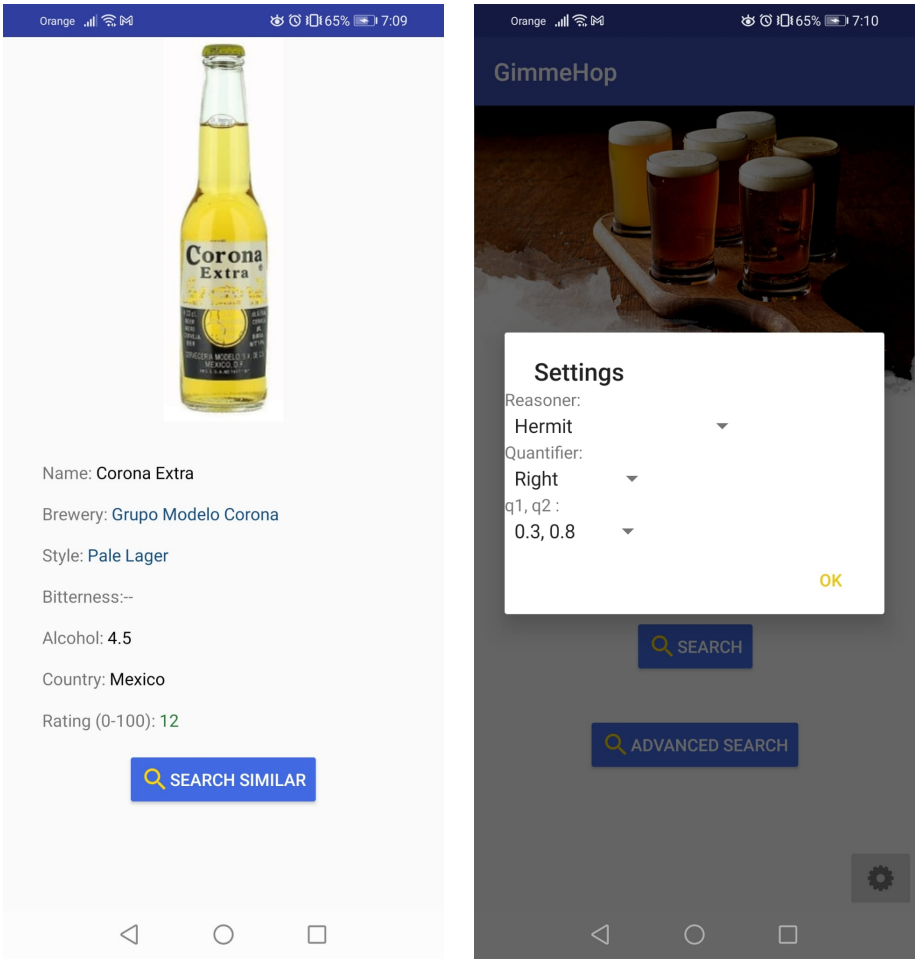
a) b)

Fig. 1. (a) GimmeHop's advanced search form, and (b) result of an advanced search

- *It manages weights internally*. To aggregate the information, the user does not need to specify the numerical values of the weights. If the user selects a feature as the most important one, the system automatically gives a higher weight to this feature. Otherwise, the user can just choose a fuzzy quantifier instead of the weight values.
- *It supports user preferences.* Users can select the relative importance of the different criteria, the reasoning mode, the reasoner (HermiT or TrOWL), the fuzzy quantifier type (right-shoulder, linear, or power), and its parameters. Figure 2 (b) shows some user settings.
- *It uses Android services for long running operations* such as loading a large ontology, classifying it, and computing the queries. Using a service is faster than using a local thread.

c) AI techniques: GimmeHop uses several Artificial Intelligence techniques:

- Fuzzy logic [4], an extension of classical logic which can represent approximate knowledge and perform approximate reasoning. GimmeHop uses:
  - Fuzzy sets, to describe linguistic labels (such as *LowAlcohol*) used to express flexible queries.
  - Fuzzy aggregation (OWA and weighted sum) [5], to combine the scores of the different criteria. Using quantifier-based OWA [6] the user does not need to choose the precise value of the weights, only the definition of the quantifier. Furthermore, one can reason with incomplete knowledge by adapting the number of weights to be computed using quantifier-based OWA to the number of available criteria.
  - Fuzzy hedges, to adapt the scoring to the user context: matching degrees of beers from the current country are increased, and matching degrees of other beers are weakened. For instance, we use the popular square root function as an increasing modifier.
- Ontologies [7], a de-facto standard for knowledge representation and reasoning. They make it possible to represent the vocabulary of a domain in a formal way, making knowledge maintenance and reuse easier. Besides, using a semantic reasoner it is possible to discover implicit



a) b)

Fig. 2. (a) Detailed information about a beer, and (b) app settings

knowledge in an automatic way. To be more precise, we use fuzzy ontologies [8], an extension of classical ontologies to represent and reason with imprecise knowledge.

- Machine learning, to compute the definitions of the linguistic labels from real data. GimmeHop uses Datil [9] to apply some centroid-based clustering algorithms and then use the centroids as the parameters of the fuzzy membership functions of the fuzzy sets. In particular, it supports k-means [10], fuzzy c-means [11], and mean-shift [12].

d) Discussion: We performed an extensive evaluation of the system (including data traffic, running time, and quality of the recommendations) on real devices. Remote reasoning is feasible and efficient in terms of data traffic and time. Local reasoning is feasible if we limit to 3000 beers. TrOWL reasoner was faster solving the queries.

We also evaluated the quality of the linguistic labels, automatically computed using the 3 supported clustering methods, with the help of 46 human aficionados. The results showed that mean-shift clustering method gave the closest results to human answers.

Our main contributions to these AI techniques are the following ones:

- A novel reasoning algorithm for (some types of) fuzzy ontologies which is able to reuse classical semantic reasoners [1], [13].
- A novel algorithm to learn the fuzzy membership functions of fuzzy ontology datatypes from numerical data [9].
- A proof of concept showing that fuzzy logic, semantic technologies, and both local and remote reasoning can be combined in mobile applications. In particular, the need to manage knowledge and reason with it on mobile devices has been recently acknowledged [3], [14].

e) Example: To clarify how GimmeHop solves advanced queries, let us consider the query illustrated in Figure 1, where the user is asking for a beer with Pale Lager style and low alcohol. We will consider several cases: the most important property can be the ABV or can be unknown, and the user location can be México (MX) or unknown.

TABLE I
RESULTS FOR TWO SAMPLE QUERIES

Beer	Original values			Partial degrees		Global degree	
	ABV	Rating	Country	ABV	Rating	Country=?	Country=MX
Query: Style=PaleLager, ABV=Low, Important property=**Alcohol** (WMEAN)							
Corona Extra	4.5	12	MX	0.59	0.12	0.405 (11th)	0.636 (1st)
Ámbar Especial	5.2	71	ES	0.38	0.71	0.519 (1st)	0.519 (2nd)
Amstel	5	59	ES	0.45	0.59	0.506 (2nd)	0.506 (3rd)
Carling	4	10	GB	0.73	0.1	0.483 (3rd)	0.483 (4th)
Asahi Super Dry	5	52	JP	0.45	0.52	0.478 (4th)	0.478 (5th)
Mahou Clásica	4.8	39	ES	0.50	0.39	0.460 (5th)	0.460 (6th)
Heineken	5	47	NL	0.45	0.47	0.458 (6th)	0.458 (7th)
Quilmes Cristal	4.9	34	AR	0.47	0.34	0.423 (7th)	0.423 (8th)
Bud Light	4.2	1	US	0.68	0.01	0.413 (8th)	0.413 (9th)
Mahou Cinco Estrellas	5.5	57	ES	0.33	0.57	0.411 (9th)	0.411 (10th)
Cruzcampo Premium Lager	5	34	ES	0.44	0.34	0.406 (10th)	0.406 (11th)
Budweiser	5	3	US	0.44	0.03	0.282 (12th)	0.282 (12th)
Query: Style=PaleLager, ABV=Low, Important property=**Indifferent** (OWA)							
Corona Extra	4.5	12	MX	0.59	0.12	0.309 (10th)	0.556 (1st)
Ámbar Especial	5.2	71	ES	0.3	0.71	0.519 (1st)	0.519 (2nd)
Amstel	5	59	ES	0.45	0.59	0.506 (2nd)	0.506 (3rd)
Asahi Super Dry	5	52	JP	0.45	0.52	0.478 (3rd)	0.478 (4th)
Heineken	5	47	NL	0.45	0.47	0.458 (4th)	0.458 (5th)
Mahou Cinco Estrellas	5.5	57	ES	0.50	0.39	0.437 (5th)	0.437 (6th)
Mahou Clásica	4.8	39	ES	0.33	0.57	0.411 (6th)	0.411 (7th)
Quilmes Cristal	4.9	34	AR	0.47	0.34	0.395 (7th)	0.395 (8th)
Cruzcampo Premium Lager	5	34	ES	0.44	0.34	0.384 (8th)	0.384 (9th)
Carling	4	10	GB	0.73	0.1	0.356 (9th)	0.356 (10th)
Bud Light	4.2	1	US	0.68	0.01	0.310 (11th)	0.278 (11th)
Budweiser	5	3	US	0.44	0.03	0.198 (12th)	0.198 (12th)

A semantic reasoner retrieves all instances (beers) of the Pale Lager class. For each of them, it retrieves its ABV, rating, and country. Then, it computes the partial degrees to be combined: the compatibility between the actual ABV value and the linguistic label *LowAlcohol*, and a normalized rating in $[0, 1]$. Finally, a global degree is computed using WMEAN (if the most important property is selected) or OWA (otherwise).

Table I shows the results (degrees and relative order). We can see that when the location is known, Mexican beers have a higher score, and indeed Corona Extra appears in the first position for both WMEAN and OWA aggregation operators.

f) Demo and installing: The interested reader can find a video demo and instructions to install the app at http://webdiis.unizar.es/~ihvdis/GimmeHop_app.

REFERENCES

[1] I. Huitzil, F. Alegre, and F. Bobillo, "GimmeHop: A recommender system for mobile devices using ontology reasoners and fuzzy logic," *Fuzzy Sets and Systems*, vol. 401, pp. 55–77, 2020.

[2] F. Bobillo and U. Straccia, "Fuzzy ontology representation using OWL 2," *International Journal of Approximate Reasoning*, vol. 52, no. 7, pp. 1073–1094, 2011.

[3] C. Bobed, R. Yus, F. Bobillo, and E. Mena, "Semantic reasoning on mobile devices: Do androids dream of efficient reasoners?" *Journal of Web Semantics*, vol. 35, no. 4, pp. 167–183, 2015.

[4] L. A. Zadeh, "Fuzzy sets," *Information and Control*, vol. 8, no. 3, pp. 338–353, 1965.

[5] G. Beliakov, H. Bustince, and T. Calvo, *A practical guide to averaging functions*, ser. Studies in Fuzziness and Soft Computing. Springer, 2016, vol. 329.

[6] R. R. Yager, "Quantifier guided aggregation using OWA operators," *International Journal of Intelligent Systems*, vol. 11, no. 1, pp. 49–73, 1996.

[7] S. Staab and R. Studer, Eds., *Handbook on Ontologies*, ser. International Handbooks on Information Systems. Springer, 2004.

[8] U. Straccia, *Foundations of Fuzzy Logic and Semantic Web Languages*, ser. CRC Studies in Informatics Series. Chapman & Hall, 2013.

[9] I. Huitzil, U. Straccia, N. Díaz-Rodríguez, and F. Bobillo, "Datil: Learning fuzzy ontology datatypes," in *Proceedings of the 17th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU 2018), Part II*, ser. Communications in Computer and Information Science, vol. 854. Springer, June 2018, pp. 100–112.

[10] S. P. Lloyd., "Least squares quantization in PCM," *IEEE Transactions on Information Theory*, vol. 28, no. 2, pp. 129–137, 1982.

[11] J. C. Bezdek, R. Ehrlich, and W. Full, "Fcm: The fuzzy c–means clustering algorithm," *Computers & Geosciences*, vol. 10, no. 2, pp. 191–203, 1984.

[12] Y. Cheng, "Mean shift, mode seeking, and clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 8, pp. 790–799, 1995.

[13] I. Huitzil, M. Molina-Solana, J. Gómez-Romero, , and F. Bobillo, "Minimalistic fuzzy ontology reasoning: An application to building information modeling," *Applied Soft Computing*, vol. 103, p. 107158, 2021.

[14] C. Bobed, F. Bobillo, E. Mena, and J. Z. Pan, "On serializable incremental semantic reasoners," in *Proceedings of the 9th International Conference on Knowledge Capture (K-CAP 2017)*. ACM, December 2017, pp. 187–190.

CAEPIA-APP Competition: Jizt - AI Summarization

Diego Miguel Lozano
Universidad de Burgos
Burgos, España
dml1001@alu.ubu.es

Carlos López Nozal
Universidad de Burgos
Burgos, España
clopzeno@ubu.es

Mario Juez Gil
Universidad de Burgos
Burgos, España
mariojg@ubu.es

***Resumen*—Jizt es una aplicación multiplataforma (Android y web) que permite realizar resúmenes abstractivos de textos y documentos. Los resúmenes abstractivos se diferencian de los resúmenes extractivos en el hecho de que no se limitan a extraer frases literales del texto original, sino que incluyen palabras o expresiones nuevas. La arquitectura de Jizt está pensada para poder acomodar diferentes modelos pre-entrenados de NLP, a fin de mantenerse siempre al día de los últimos avances en el campo. En este trabajo se presenta la funcionalidad y utilidad de Jizt y se describe en detalle los componentes de su arquitectura software. Además, se presenta el diseño del algoritmo de balanceo de texto necesario como fase de preprocesado textual antes de la generación del resumen.**

***Descriptores*—NLP, generación de resúmenes abstractivos, Cloud Native, arquitectura de microservicios, aplicación multiplataforma**

I. Introducción

Uno de los campos históricos dentro de la Inteligencia Artificial (IA o AI) es el Procesamiento del Lenguaje Natural (NLP, por sus siglas en inglés), cuya significación se hizo patente con la aparición del célebre Test de Turing [1], en el cual un interrogador debe discernir entre un humano y una máquina conversando con ambos por escrito a través de una terminal.

Hasta los años 80, la mayor parte de los sistemas de NLP estaban basados en complejas reglas escritas a mano [2], las cuales generalmente daban lugar a modelos muy lentos, poco flexibles y con baja precisión. A partir de esta década, como fruto de los avances en Aprendizaje Automático (*Machine Learning)*, fueron apareciendo modelos estadísticos, consiguiendo notables avances en campos como el de la traducción automática.

En la última década, el desarrollo ha sido aún mayor debido a factores como el aumento masivo de datos de entrenamiento (principalmente provenientes del contenido generado en la web), avances en la capacidad de computación (GPU, TPU, ASIC...) y el progreso dentro del área de la Algoritmia [3].

No obstante, ha sido desde la aparición del concepto de «atención» en 2015 [4]–[6] cuando el campo del NLP ha comenzado a lograr resultados cuanto menos sorprendentes [7], [8].

Con todo, la mayor parte de estos avances se han visto limitados al ámbito académico y corporativo. Los modelos cuyo código ha sido publicado, o bien no están entrenados, o bien requieren conocimientos avanzados para poder ser usados.

Con esta idea en mente, el objetivo de Jizt se centra en acercar los modelos NLP preentrenados del estado del arte (*Transformers*) tanto a usuarios expertos, como no expertos. Para ello, Jizt proporciona:

- Una plataforma en la nube que expone una API REST a través de la cual se puede solicitar la generación de resúmenes. Esta opción está dirigida a usuarios con conocimientos técnicos.
- Una aplicación multiplataforma que consume dicha API, y que proporciona una interfaz gráfica sencilla e intuitiva. Esta aplicación está pensada para ser utilizada por el público general.

El desarrollo de Jizt ha sido financiado parcialmente gracias a la Beca de Colaboración en Departamentos Universitarios del Ministerio de Educación y Formación Profesional del Gobierno de España.

II. Recursos online

- Página web oficial: *www.jizt.it*.
- Aplicación web: *app.jizt.it*.
- Aplicación Android (Google Play): *https://play.google.com/store/apps/details?id=it.jizt.app*.
- Documentación en inglés: *docs.jizt.it*.
- Documentación en español (más completa pero algo menos actualizada): *dmlls.github.io/jizt-tfg*.
- Código fuente: *github.com/jizt-it*.
- Vídeo demostrativo: *www.jizt.it/videos/caepia-2021/*.

III. Descripción de la aplicación

Actualmente, para la generación de resúmenes, Jizt hace uso del modelo T5 [9], desarrollado por Google en 2019. Los modelos de NLP del estado del arte, como el T5, requieren de grandes capacidades de computación. Es por ello que por el momento no son aptos para ser ejecutados en un dispositivo móvil. Por tanto, en nuestro caso, el tratamiento de los textos y la generación de los resúmenes se traslada a la nube, donde contamos con una infraestructura capaz de llevar a cabo estas tareas, consiguiendo que nuestra aplicación móvil tan solo ocupe 6,3 *megabytes*, en comparación con los varios *gigabytes* que pueden llegar a ocupar los modelos de NLP.

III-A. Importancia de la generación de resúmenes

En la actualidad, existen multitud de aplicaciones y servicios tanto de pago como gratuitos que llevan a cabo diversas tareas de NLP, como *bots* conversacionales, traducción automática,

búsqueda de respuestas, etc. Detrás de gran parte de este tipo de servicios se encuentran gigantes como Google con, por ejemplo, Google Assistant [10], Google Translate [11]; Amazon con Alexa [12], Amazon Comprehend [13]; o IBM con Watson [14], etc. Asimismo, se pueden encontrar multitud de proyectos *open-source*. No obstante, a día de hoy existen pocas opciones en lo que a la generación de resúmenes abstractivos se refiere. Es ahí donde nace Jizt para cubrir ese nicho de mercado.

La generación de resúmenes podría ser de utilidad en múltiples áreas relacionadas con la trasmisión de información textual como: educación, investigación, publicidad, creación de noticias y de contenidos textuales en redes sociales, etc. En general, cualquier empresa que trabaje con datos textuales podría beneficiarse de ello, pudiéndoles ofrecer la información más importante a "golpe de vista", ahorrando tiempo y esfuerzo, y facilitando la búsqueda de información. Por otro lado, también se podría aplicar como punto de partida para otras tareas de NLP, como búsqueda de respuestas o recuperación y extracción de información.

III-B. Funcionalidad principal de la app

La aplicación de Jizt se encuentra en activo desarrollo, incluyendo en cada iteración nuevas funcionalidades y mejoras.

Por el momento, la aplicación permite generar resúmenes introduciendo directamente el texto a resumir, así como a partir de documentos (PDF, EPUB, Microsoft Word...).

El usuario puede seleccionar la longitud mínima y máxima del resumen de manera relativa a la longitud del texto original.

Una vez generado el resumen, el usuario puede compartirlo de manera sencilla en otras plataformas o aplicaciones, directamente desde la *app* de Jizt.

Los resúmenes generados se almacenan de manera local en el dispositivo, por lo que se pueden volver a visualizar en cualquier momento, sin necesidad de generarlos de nuevo.

En cualquier caso, el sistema cuenta con una capa de caché tanto a nivel local como a nivel remoto, por lo que si un texto ya ha sido resumido previamente, el sistema devolverá el resumen de manera instantánea. A fin de respetar la privacidad de nuestros usuarios, y dado que los textos introducidos podrían tener carácter privado, ofrecemos al usuario la posibilidad de no *cachear* sus textos, de forma que, una vez ha sido generado el resumen, tanto el texto original como el resumen son eliminados de nuestras bases de datos de manera permanente.

III-C. Herramientas utilizadas

Jizt se divide en dos partes fundamentales:

- *Backend*: se encarga del procesamiento de los textos y la generación de los resúmenes. Sigue una arquitectura de microservicios dirigida por eventos, lo que garantiza la escalabilidad, alta disponibilidad y la tolerancia frente a fallos del sistema.
- *Frontend*: se encarga de la interacción con el usuario, proporcionando una interfaz sencilla e intuitiva. El *frontend* se ofrece por el momento como aplicación para Android, así como para web.

La infraestructura del *backend* actualmente está desplegada en Google Kubernetes Engine (GKE) [15], un servicio de Google Cloud. Para la comunicación entre microservicios se ha empleado Apache Kafka [16], un *framework* que destaca por su gran capacidad de escalado, su funcionamiento distribuido, y su tolerancia frente a fallos. Para el almacenamiento de los resúmenes se hace uso de PostgreSQL [17]. La generación de los resúmenes se lleva a cabo gracias al *framework* "transformers" ofrecido por Hugging Face [18], que proporciona asimismo los modelos de NLP preentrenados.

El *frontend* se ha desarrollado con Flutter [19], lo que nos permite compilar nuestra aplicación al código nativo de las diferentes plataformas existentes: móvil (Android e iOS), web, y escritorio (Linux, Windows e iOS). Hasta el momento, el trabajo se ha centrado en dar soporte para Android y web, pero se prevé ampliarlo al resto de plataformas más adelante.

III-D. Utilidad y viabilidad de la app

Uno de los grandes desafíos hoy en día reside en el traslado de los avances y nuevos modelos de NLP a productos o herramientas que el público general pueda utilizar.

Como se menciona anteriormente, el gran tamaño de los modelos impide que puedan ser ejecutados directamente sobre dispositivos como *smartphones* u ordenadores convencionales.

Al mismo tiempo, el desarrollo de una infraestructura en la nube que permita a los usuarios acceder a dichos modelos presenta de igual modo grandes dificultades, como por ejemplo el hecho de que múltiples usuarios deberían, en principio, ser capaces de hacer uso del servicio simultáneamente.

Todos estos aspectos fueron considerados desde las primeras fases de desarrollo de Jizt, dando lugar a una infraestructura robusta y plenamente funcional.

En cuanto a la viabilidad del proyecto, los costes provienen principalmente de la contratación de los servicios *cloud* para desplegar el *backend*. Actualmente, estamos trabajando en poner en marcha un modelo de financiación centrado en el uso de nuestra API REST. Esencialmente, todo usuario podrá generar cierto número de resúmenes de manera gratuita. No obstante, aquellos que requieran de un uso más intensivo del servicio tendrán que abonar cierta cantidad por ello, en función del uso. También ofreceremos a aquellas empresas o particulares interesados, la posibilidad de llevar a cabo un despliegue propio del *backend* de Jizt, bien en sus propias dependencias, o bien a través de un *cloud provider*. Este tipo de clientes tendrán control total sobre el *backend*, sin experimentar ningún tipo de limitación.

IV. Algoritmos y técnicas de IA empleadas

El proceso de generación de resúmenes se divide en cuatro etapas fundamentales:

1. Pre-procesado: en esta etapa, se realizan pequeñas modificaciones sobre el texto de entrada para adecuarlo a la entrada que el modelo de generación de resúmenes espera.



Figura 1. Ejemplo gráfico del algoritmo de balanceo. Una vez dividido el texto vorazmente, procedemos a equilibrar el número de *tókenes* en cada fragmento. En este caso, suponemos un hipotético tamaño máximo de entrada de 100 *tókenes*. Las desviación estándar del número de *tókenes* de cada frase en t_1 es $\sigma_1 = 39,63$ y en t_5 acaba siendo $\sigma_5 = 1,53$.

2. Codificación: a través del *tokenizador*, el cual es parte del modelo T5, se divide el texto en *tókenes*[1] y se *mapean* a vectores numéricos, con los cuales puede trabajar el generador de resúmenes.
3. Generación del resumen: en esta etapa es cuando se produce la generación del resumen propiamente dicha.
4. Post-procesado: el modelo T5 produce textos únicamente en minúsculas. En esta etapa, se recomponen las mayúsculas allí donde son necesarias. Para ello utilizamos el método propuesto en [20].

Los modelos pre-entrenados de generación del estado del arte presentan frecuentemente limitaciones en la longitud de los textos de entrada que reciben, derivadas de la longitud de las secuencias de entrada con las que han sido entrenados. Esta longitud llega a ser tan baja como 512 *tókenes*.

Dos estrategias comunes para eludir esta limitación consisten en: (a) truncar el texto de entrada, lo cual puede dar lugar a pérdidas notables de información, o (b) dividir el texto en fragmentos de menor tamaño. En el caso de Jizt, la primera opción queda rápidamente descartada ya que los textos que recibimos, por lo general, superan el tamaño máximo.

Por tanto, la manera en la que se procede es la siguiente:

1. División del texto en fragmentos.
2. Generación de un resumen de cada fragmento.
3. Concatenación los resúmenes parciales.

En cuanto a la división del texto, es frecuente llevar a cabo dicha separación de manera "ingenua", únicamente atendiendo al tamaño de entrada máximo, a través de un algoritmo voraz que va completando el fragmento actual hasta llegar al tamaño máximo, y a continuación, comienza a completar el siguiente fragmento.

Un potencial problema con este modo de proceder reside en que el último de los fragmentos podría contener un número arbitrariamente pequeño de palabras. En este caso, el modelo produciría una salida incompleta o simplemente vacía.

El segundo problema radica en que, si se atiende únicamente al número de *tókenes*, es altamente probable que una frase quede segmentada entre dos fragmentos diferentes, de nuevo pudiendo dar lugar a resúmenes que no recogen algunos de los aspectos importantes del texto de entrada.

La forma en la que se procede, a fin de resolver estos dos problemas, consiste en:

1. Identificar las frases en el texto y tratarlas como unidad indivisible a la hora de realizar la fragmentación.
2. Asegurar que la variación en el número de *tókenes* entre los diferentes fragmentos sea lo más pequeña posible.

Para la detección de frases se emplea la librería Bling Fire [21] combinada con expresiones regulares para resolver casos en los que dicha librería falla, por ejemplo, fechas en el siguiente formato `01.02.1990`, en las que el modelo identificaría el fin de una frase y el comienzo de otra.

En cuanto a la regularización del número de *tókenes* en cada fragmento, se ha desarrollado un algoritmo propio que se divide en dos etapas:

1. Dividir el texto vorazmente.
2. Equilibrar el número de *tókenes* en cada fragmento.

El proceso de equilibrado una vez dividido el texto vorazmente se muestra gráficamente en la Figura 1, y su formula-

[1]Para modelos que trabajan a nivel de sub-palabra, como el caso del T5, un *token* es bien una palabra, o bien un fragmento de la misma.

ción en pseudocódigo se puede encontrar en el Algoritmo 1.

En esencia, lo que este último algoritmo hace es comparar la diferencia en número de *tókenes* entre fragmentos consecutivos, empezando por el final, de forma que primero se compara el penúltimo fragmento con el último, después el antepenúltimo con el penúltimo, y así sucesivamente. Si es necesario, se van trasladando frases completas desde un fragmento al siguiente, por ejemplo, desde el penúltimo al último fragmento. Este algoritmo tiene una complejidad en el peor de los casos de $O(n^3)$, siendo n el número de fragmentos.

En [22] se describe el algoritmo completo de manera más detallada.

Algoritmo 1 Balanceo de las fragmentos.

```
1: procedure BALANCEAFRAGMENTOS(ptosCorte)
2:     ptosCorteBalan ← ptosCorte                ▷ Puntos de corte balanceados
3:     do
4:         ptosCorteBalanOld ← ptosCorteBalan
5:         for i ← len(ptosCorteBalan) − 1, 1, step : −1 do   ▷ Empezar por último fragmento.
6:             diffLen ← lenFrag(i -1) − lenFrag(i)   ▷ Diferencia en n. de tókenes
7:             while diffLen > 0 do
8:                 últimaFrase ← getÚltimaFrase(getFrag(i-1))
9:                 if getFrag(i) + len(últimaFrase) ≤ model.maxLength ∧
                          len(últimaFrase) ≤ diffLen then
10:                    mueveÚltimaFrase(getFrag(i-1), getFrag(i))
11:                    ptosCorteBalan[i − 1] ← ptosCorteBalan[i − 1] − 1
12:                else
13:                    break
14:                end if
15:            end while
16:        end for
17:    while ptosCorteBalan ≠ ptosCorteBalanOld
18:    return ptosCorte
19: end procedure
```

V. Conclusiones

En este trabajo se presenta la aplicación multiplataforma Jizt, que permite realizar resúmenes abstractivos de textos y documentos a partir de modelos pre-entrenados de NLP. Además de la funcionalidad y utilidad de Jizt, se describen en detalle los componentes de su arquitectura *software* orientados a solucionar los problemas de diseño de este tipo de aplicaciones de NLP. Algunos de los problemas abordados están relacionados con la eficiencia, escalabilidad y privacidad de los datos textuales a resumir proporcionados por un usuario. Nuestro diseño *software* basado en *framework* resuelve los problemas de extensión a nuevos modelos de NLP y su parametrización específica.

Desde un punto de vista más orientado a la tarea de generación de resúmenes de NLP, se presenta el diseño de un algoritmo de balanceo de texto, necesario como fase de preprocesado textual antes de la generación del resumen. Empíricamente, se ha observado que la aplicación de este algoritmo de balanceo puede mejorar, de forma general, la calidad de los resúmenes generados.

Referencias

[1] A. M. Turing. *Computing Machinery and Intelligence*. En: MindLIX.236 (oct. de 1950), págs. 433-460.

[2] Pamela McCorduck. *Machines Who Think*. USA: W. H. Freemany Co., 1979. ISBN: 0716710722.

[3] Joachim Rahmfeld. *Recent Advances in Natural Language Processing*. Sep. de 2019. URL: https://damapdx.org/2019/08/23/september-2019-recent-advances-in-natural-language-processing-some-theory-applications. Último acceso: 16-06-2021.

[4] Ashish Vaswani y col. *Attention Is All You Need*. En: CoRRabs/1706.03762 (2017). arXiv: 1706.03762.

[5] Thang Luong, Hieu Pham y Christopher D. Manning. *Effective Approaches to Attention-based Neural Machine Translation*. En: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. Lisbon, Portugal: Association for Computational Linguistics, sep. de 2015, págs. 1412-1421.

[6] Dzmitry Bahdanau, Kyunghyun Cho y Yoshua Bengio. *Neural Machine Translation by Jointly Learning to Align and Translate*. 2016. arXiv: 1409.0473 [cs.CL].

[7] Thomas Macaulay. *Someone let a GPT-3 bot loose on Reddit — it didn't end well*. Oct. de 2020. URL: https://thenextweb.com/neural/2020/10/07/someone-let-a-gpt-3-bot-loose-on-reddit-it-didnt-end-well. Último acceso: 16-06-2021.

[8] Kyle Wiggers. *AI models from Microsoft and Google already surpass human performance on the SuperGLUE language benchmark*. Ene. de 2021. URL: https://venturebeat.com/2021/01/06/ai-models-from-microsoft-and-google-already-surpass-human-performance-on-the-superglue-language-benchmark. Último acceso: 16-06-2021.

[9] Colin Raffel y col. *Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer*. En: CoRRabs/1910.10683 (2019), pág. 11. arXiv: 1910.10683.

[10] Google. *Google Assistant*. Jun. de 2021. URL: https://assistant.google.com. Último acceso: 16-06-2021.

[11] Google. *Google Translate*. Jun. de 2021. URL: https://translate.google.com. Último acceso: 16-06-2021.

[12] Amazon. *Amazon Alexa*. Jun. de 2021. URL: https://alexa.amazon.com. Último acceso: 16-06-2021.

[13] Amazon. *Amazon Comprehend*. Jun. de 2021. URL: https://aws.amazon.com/comprehend. Último acceso: 16-06-2021.

[14] IBM. *IBM Watson*. Jun. de 2021. URL: https://www.ibm.com/watson. Último acceso: 16-06-2021.

[15] Google Cloud. *Google Kubernetes Engine*. Jun. de 2021. URL: https://cloud.google.com/kubernetes-engine. Último acceso: 16-06-2021.

[16] Apache Software Foundation. *Apache Kafka*. Jun. de 2021. URL: https://kafka.apache.org. Último acceso: 16-06-2021.

[17] PostgreSQL Global Development Group. *PostgreSQL*. Jun. de 2021. URL: https://www.postgresql.org. Último acceso: 16-06-2021.

[18] Hugging Face. *Transformers*. Jun. de 2021. https://github.com/huggingface/transformers. Último acceso: 16-06-2021.

[19] Flutter. *Flutter - Hermosas apps nativas en tiempo récord*. Jun. de 2021. URL: https://esflutter.dev. Último acceso: 16-06-2021.

[20] Lucian Vlad Lita y col. "TRuEcasIng". En: Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1. ACL '03. Sapporo, Japan: Association for Computational Linguistics, 2003, págs. 152-159. DOI: 10.3115/1075096.

[21] Microsoft. *Bling Fire*. Mar. de 2021. URL: https://github.com/microsoft/BlingFire. Último acceso: 16-06-2021.

[22] Diego Miguel. *Jizt - Conceptos teóricos*. Feb. 2021. URL: https://dmlls.github.io/jizt-tfg/memoria/3_Conceptos_teoricos.html#codificacion-del-texto. Último acceso: 16-06-2021.

Competición CAEPIA-App: KSAS: A Mobile App with Neural Networks to Guide the Learning of Motor Skills

Alberto Casas-Ortiz
Artificial Intelligence Dept.
Computer Science School, UNED
Madrid, España
acasas148@alumno.uned.es

Olga C. Santos
aDeNu Research Group. Artificial Intelligence Dept.
Computer Science School, UNED
Madrid, España
ocsantos@dia.uned.es

***Abstract*—COVID-19 pandemic has forced instructors and learners of motor skills (e.g., martial arts, dance, perform medical surgeries...) to cease their training and practice in sessions face-to-face. Video-conference systems have been used as a temporal solution as they facilitate learning from home at any time, complemented with the use of videos, but this entails some disadvantages. As a solution to this we propose KSAS, a neural network-based Android app to guide the learning of motor skills, specifically in martial arts. This mobile app is able to capture the movements of a part of the body, model them as time series using the EWMA algorithm, analyze the movements using an LSTM neural network, and provide multisensorial feedback to the learner. KSAS is available for download and can be used to train at home (either alone or on-line) as well as in a face-to-face classroom supporting the instructor.**

***Index Terms*—psychomotor learning, intelligent tutoring systems, motor skills, artificial intelligence, neural networks, martial arts, feedback, sensors**

I. Introduction

Learning a motor skill requires the coordination of different parts of the body with the purpose of accomplishing a physical task [1]. We can find examples of motor skills everywhere (e.g., martial arts, dance, play musical instruments, medical surgery...). Traditionally, the learning of motor skills has been done face-to face, from an instructor to a learner, and it is usually done in a specialized training center.

Situations like the outbreaks of the COVID-19 pandemic can force instructors and learners of motor skills to stop their training, ceasing their practice sessions face-to-face. Video-conference systems have been used as a temporal solution as they facilitate learning from home at any time complemented with the use of videos that can be paused or rewound. However, this entails some disadvantages. In a questionnaire answered by 29 learners and instructors of psychomotor activities [3], we identified the following problems: i) technical problems derived from the internet connection during live sessions, ii) lack of interaction with instructors and learners, iii) lack of feedback from the instructor, and iv) difficulty to observe 3D movements in 2D displays.

With the purpose of attenuating those disadvantages, in our research we use smartphones as wearables. Smartphones can be used for capturing, modeling and analyzing 3D inertial information. Since a smartphone is in essence a computer, it can be used for modeling and analyzing human movements using Artificial Intelligence (AI) techniques to personalize the learning process, and some components (e.g., vibrators, display and speakers) can be used for providing personalized multimodal feedback. Also, smartphones can be used to share information online, allowing the teacher to visualize the learner's progress and thus, encouraging the interaction between them.

To expose the advantages that the use of smartphones entails when learning motor skills online, we have developed KSAS[1] (Kenpo Set Assisting System), an AI assisted mobile application developed in Android to learn motor skills in online settings. To cover the lack of standards and methodologies in the development of systems and applications to assist the learning of motor skills, we have designed the framework mCMARR, that allows to develop applications to learn motor skills in a modular and simple way.

This paper is organized as follows. Section 2 performs a brief review of the state of the art in the use of smartphones as wearables when learning motor skills. Section 3 describes the proposed framework for developing applications and systems to learn motor skills: mCMARR. Section 4 presents and describes the application KSAS, and its implications when learning martial arts movements. Finally section 5 presents our conclusions.

II. State of the art

The development of applications and systems to assist the learning of psychomotor activities is a young field of research [2]. Different bibliographical reviews have been done with the purpose of identifying available technological support

The work reported is framed in the project "INTelligent INTra-subject approach to improve actions in AFFect-aware adaptive educational systems" (INT2AFF) funded under Grant PGC2018-102279-B-I00 (MCIU/AEI/FEDER, UE).

[1] https://blogs.uned.es/phyum/ksas/ (app and video available from here)

to build systems for learning motor skills [2]–[6]. The use of smartphones as wearables to capture, model and analyze motion with the purpose of learning motor skills have been proposed elsewhere [4], however, only one of the systems identified in that paper (i.e. [7]) uses smartphones as wearables to assess the learning of motor skills.

With the purpose of identifying other systems using smartphones as wearables to asses the learning of motor skills, we have performed a new bibliographical review following the same methodology as in [5], but focused on mobile-based systems. The databases explored, the search query applied and the results obtained can be seen in Table I.

As Table I shows, only [7] has been found (already found in [4]). In [7], a smartphone is used as wearable, located in a pocket, to capture inertial information to assist the learning of social dancing. Visual feedback is given after the practice of the movements in the form of charts, summaries and texts.

We have also researched commercial mobile apps that can support the learning of motor skills by capturing human motion, but no relevant results have been found either in the application stores of Google Play and App Store.

TABLE I
INFORMATION ABOUT THE REVIEW OF THE STATE OF THE ART

Search Query	(Motor Skills OR Psychomotor Activity OR Psychomotor Learning OR Psychomotor Domain) AND (Smartphone OR Mobile Device OR Android OR Iphone) AND (Wearable)
Inclusion Criteria	- Is a full paper or a conference proceeding - A smartphone is used as wearable - The smartphone is used to capture motion and give feedback - The application is used to learn a motor skill
Databases Explored	IEEE Xplore (7 results), PubMed (7 results), ACM DL (66 results)
Relevant Results	[7]

III. MCMARR FRAMEWORK

With the purpose of easing the development of systems and applications to learn motor skills, we propose the framework mCMARR, that described the flow of information divided in five modules: i) **m**otion **C**apture, ii) **m**otion **M**odel, iii) **m**otion **A**nalysis, iv) **m**otion **R**esponse and v) **m**otion **R**eport. This framework is based on the ideas of [2], and can be used to develop and understand any kind of application and system that assist the learning of motor skills.

The diagram in Figure 1 shows the flow of information of the framework. First the Motion Response module gives some initial indications to the learner, who has to execute the movements following the indications. The executed movements are captured by the Motion Capture module, and the raw captured data is modelled in the Motion Model module. The raw captured data and the modeled data is then analyzed in the Motion Analysis module, which extracts useful information about the performance of the movements. The extracted information is then used by the Motion Response module to give feedback and more indications to the learner, as well as to give information to the Motion Reports module. The Motion Reports module then generates a report that can be shared with the teacher, the learner, and the learner's classmates to follow the progress during the learning. Using the generated reports, the teacher can then decide if the learner needs further intervention during the learning process.

IV. KSAS: AN APP TO GUIDE THE LEARNING OF MOTOR SKILLS

KSAS (Kenpo Set Assisting System) [8], is an application that guides the learning of a set of movements from American Kenpo Karate [9]. We have selected martial arts as the motor skill studied in this research since it encompasses many of the characteristics common to other motor skills [6] like: i) interaction with other people (e.g., executing a self defense technique against an opponent), ii) coordination of different parts of the body (e.g., execution of sequences of simultaneous movements in katas), iii) use of specialized items (e.g., use of weapons) and iv) response to external stimuli (e.g., response to an attack executed by an opponent). Also, the learning of a martial art is usually guided in the first stages of learning.

Since the use of a smartphone as wearable limits the capture of the movements to the part of the body where the mobile is placed, the selected movements to be learned by using KSAS must be executed by only one part of the body at the same time. This makes the set of movements known as Blocking Set I perfect for this application, since the first part of the set consists in executing blocking movements with each arm separately. Additionally, the most common error executed by beginners while learning the Blocking Set I is the execution of the movements in the incorrect order, so KSAS focuses in teaching the correct order of execution of the movements. The five movements that are learned using the application are: upward block, inward block, outward block, outward downward block and rear elbow block.

The development of the application KSAS has been done in Android (XiaoMi Mi2), by following the mCMARR framework. The current version extends the outcomes of the Master Thesis [3], which received a mention in eMadrid Awards[2], and the improved version presented in the Interactive Events of the AIED conference[3] [8]. KSAS application has also been presented at Madrid Science and Innovation Week in Nov. 2020[4] and in a talk about "Sports and AI"[5] at the IA Center of Ourense in Spain.

The following subsections show how each module of the framework has been implemented in the application.

A. Capturing human motion

Since we are using a smartphone as a wearable located in the arm, we are capturing human motion by reading information from the inertial sensors of the device. Six signals have been captured in the three spatial axes at a approximate rate of 20

[2]https://www.emadridnet.org/
[3]https://iaied.org/showcase
[4]http://www.madrimasd.org/semanacienciaeinnovacion/
[5]https://intelixencia-artificial.ourense.gal/

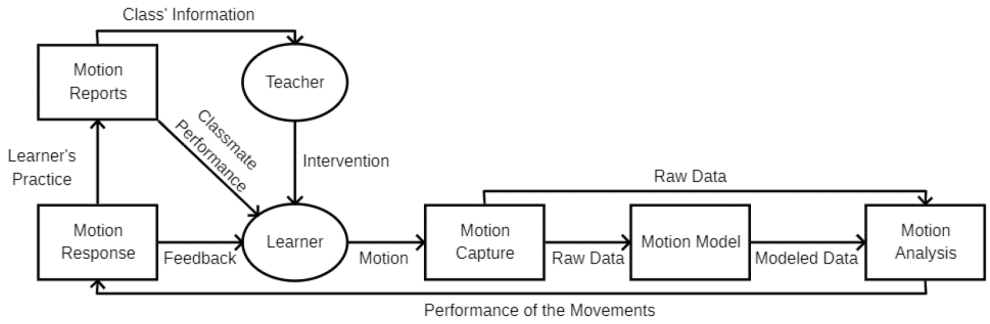


Fig. 1. Flow of information and modules of the mCMARR framework.

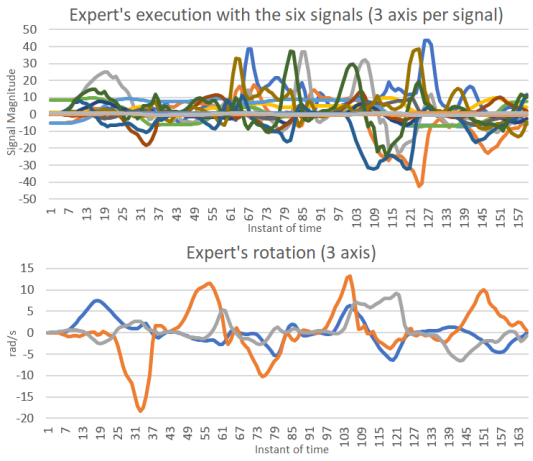


Fig. 2. Representation of the five captured movements as a time series. The top chart represents the execution of the five movements by an expert, and the bottom chart represents only the gyroscope information, which seems useful to differentiate the movements.

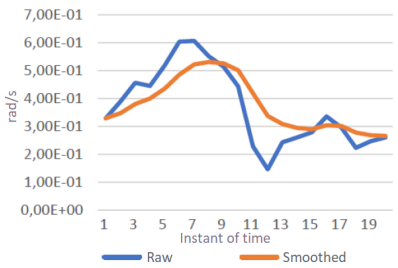


Fig. 3. Comparison between a raw signal captured by the smartphone, and the same signal smoothed using EWMA.

ms (SENSOR_DELAY_GAME). The Android sensors used as source are the following [6]: accelerometer, gravity, gyroscope, linear acceleration, game rotation vector and magnetic field.

In Figure 2, the top chart shows a representation of all the captured signals executed by an expert as a synchronized time series, and the bottom chart shows only the gyroscope information, which seems useful to differentiate the movements.

[6]https://developer.android.com/guide/topics/sensors/

B. *Modelling human motion*

The captured signals have been interpreted and modeled as time series, by synchronizing them over time (Figure 2). With the purpose of removing noise and smoothing the curves, the Exponentially Weighted Moving Averages (EWMA) [10] algorithm has been applied to each captured time series, since the quantity of smoothness applied can be tuned by a parameter β. Figure 3 shows an example of a time series smoothed using EWMA.

C. *Analyzing human motion*

The modeled information is analyzed by a Long Short-Term Memory (LSTM) neural network. This neural network has been trained over a dataset generated from the executions of the movements of 20 volunteers, with both arms. The gathering of data has followed the guidelines of the ethical committee of the UNED, and all volunteers gave their consent by signing the corresponding informed consent form.

The dataset consist in 200 captured movements (5 movements x 20 volunteers x 2 arms), plus 40 instances of noise. This gives a total of 240 instances, in which each instance is formed by 18 synchronized time series. Since there are 5 movements in the dataset and noise (lack of movement), the dataset is formed by six classes.

The LSTM neural network has been trained following a 10-fold setting. The optimizer used in the training is Adam, and the loss function used is logcosh. The value used for the learning rate was 0.0001, and the parameter β of the EWMA algorithm has been tuned, by testing the values 0.1, 0.3, 0.5, 0.7, 0.9 and 1, without producing significant differences. The dataset has been divided into a training set (70% of instances) and a test set (30% of instances). The structure of the LSTM neural network can be seen in Figure 4.

The LSTM neural network obtained always a mean of 100% of accuracy in the training set (mean calculated over the results of the 10 folds) and a mean of 89%-92% of accuracy in the testing set (the value depends on the value of β of the EWMA algorithm). A 1-Dimensional Convolutional Neural Network (1-D CNN) and a Multilayer Perceptron (MP) have been tested and compared with the LSTM neural network, but the later

outperformed both in any case, possibly due to its suitability to analyze time series.

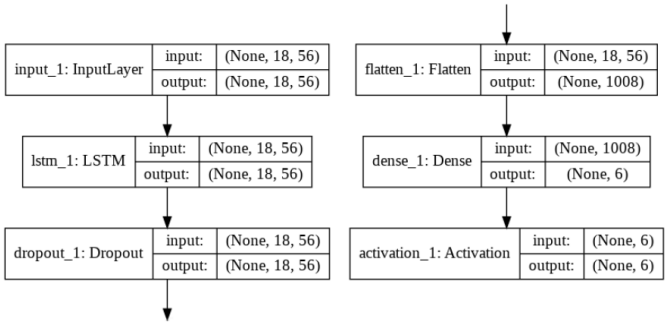


Fig. 4. Structure and layers of the LSTM neural network.

D. Giving a response or feedback

Multimodal feedback can be given by using the different components of a smartphone. For instance, visual feedback can be given by using the display of the device, auditory feedback can be given by using the speakers of the device, and haptic feedback can be given by using the vibrator of the device.

The feedback strategy implemented in KSAS is as follows. First, the application gives initial indications as verbal instructions (auditory feedback) asking the learner to execute a movement. If the execution of the movement is correct, the application plays a victory sound and congratulates the learner (auditory feedback), asking him/her to execute the next movement. If the execution of the movement is incorrect, the device vibrates (haptic feedback), and the application plays a buzz sound and asks the learner to execute the movement again (auditory feedback). All verbal instructions are given either in Spanish of English, depending of the default language of the smartphone. Visual feedback is not given by the application since the display of the device is not always visible to the learner when performing the movements.

E. Generating a report

Once the execution has finished, the application is able to count the number of errors executed by the learner. The number of errors of the current session is stored, and a report is generated with the evolution of errors over time. These reports can then be shared via internet connection with the instructor and other classmates (only if the learner consents), with the purpose of allowing the teacher to follow the progress of the learner, and encourage the training between classmates with gamification techniques.

V. Conclusion

In this document, we have addressed the problem of learning motor skills online and the problems that derive from the use of video-conference systems and videos, and we propose a solution that uses smartphones as wearables. First, to ease the development of applications and systems to learn motor skills, we have defined a framework called mCMARR formed by five modules: i) motion Capture, ii) motion Model, iii) motion Analysis, iv) motion Response and v) motion Reports. This framework has then been used to develop an AI application called KSAS to learn a set of movements of a martial art known as American Kenpo Karate. KSAS is able to capture motion using a smartphone as a wearable by reading information from the inertial sensors of the device. The captured information is modeled as a data series and smoothed using the EWMA algorithm. The modeled movements are analyzed by using a LSTM neural network to determine if the movements are executed in the correct order. Verbal instructions, sounds and vibrations are used to give feedback to the learner.

The use of smartphones allows training at any place in any moment, in a cheap and easy way. However, the fact that only one part of the body can be captured at the same time is a limitation when learning movements that requires the coordination of different parts of the body, but it is useful for learning motor skills that only involve one limb of the body (e.g., play tennis, play violin...).

The analysis of the movements using neural networks gives a good accuracy. However, it is difficult to understand which characteristics of the movements are important for the analysis, and it is difficult to generalize the analysis to other movements without training the networks again, so different techniques should be explored.

Preliminary results performed with 6 volunteers divided into three groups (blended group, videoconference group and app group) suggest that the application is useful for learning, but more volunteers are necessary to draw stronger conclusions.

References

[1] R. A. Magill and D. I. Anderson, Motor Learning and Control: Concepts and Applications, 11th ed. McGraw-Hill Education, 2017.
[2] O. C. Santos, “Training the Body: The Potential of AIED to Support Personalized Motor Skills Learning,” Int. J. Artif. Intell. Educ., vol. 26, no. 2, pp. 730–755, 2016, doi: 10.1007/s40593-016-0103-2.
[3] A. Casas-Ortiz, “Capturing, Modelling, Analyzing and providing Feedback in Martial Arts with Artificial Intelligence to support Psychomotor Learning Activities,” Master Thesis. Supervised by O.C. Santos. Universidad Nacional de Educación a Distancia, June 2020.
[4] O. C. Santos and M. H. Eddy, “Modeling Psychomotor Activity: Current Approaches and Open Issues,” UMAP’17 Adjun., pp. 305–310, 2017, doi: 10.1145/3099023.3099083.
[5] A. Casas-Ortiz and O. C. Santos, “Chapter II-8. Intelligent Systems for Psychomotor Learning,” in Handbook of Artificial Intelligence in Education, Edward Elgar Publishing. In progress.
[6] O. C. Santos, “Psychomotor Learning in Martial Arts: an Opportunity for User Modeling, Adaptation and Personalization,” Adjun. Publ. 25th Conf. User Model. Adapt. Pers. - UMAP ’17, pp. 89–92, 2017, doi: 10.1145/3099023.3099107.
[7] A. D. P. Dos Santos, K. Yacef, and R. Martinez-Maldonado, “Let’s Dance: How to build a user model for dance students using wearable technology,” 2017, doi: 10.1145/3079628.3079673.
[8] A. Casas-Ortiz and O. C. Santos, “KSAS: An AI Application to learn Martial Arts Movements in on-line Settings,” Interactive Events at AIED21 (Artificial Intelligence in Education), 2021.
[9] E. K. Parker, Infinite Insights Into Kenpo - Volumes 1-5. BookSurge Publishing, 2009.
[10] M. B. Perry, “The Exponentially Weighted Moving Average,” in Wiley Encyclopedia of Operations Research and Management Science, 2011.

Competición CAEPIA-App: Sugerencias de Moda

Ismael Pineda-Palencia, Rafael Marcos Luque-Baena, Miguel A. Molina-Cabello
Departmento de Lenguajes y Ciencias de la Computación. Universidad de Málaga, España
Instituto de Investigación Biomédica de Málaga – IBIMA, España
Emails: ismael.pineda@hotmail.com; {rmluque,miguelangel}@lcc.uma.es

***Resumen*—Sugerencias de Moda es una aplicación móvil (app) que plasma la renovación y modernización de los sistemas de búsqueda tradicionales basados en texto, gracias a los avances de la Inteligencia Artificial en el campo de la Visión por Computador. Esta aplicación tiene como principal objetivo simplificar a los usuarios el proceso de búsqueda de prendas y complementos de moda similares a los que aparezcan en las imágenes digitales que faciliten a la aplicación. Las sugerencias se realizan en tiempo real mediante la información ofrecida por modelos entrenados por medio de técnicas de Aprendizaje Profundo.**

I. Justificación de la investigación

Durante los últimos años las empresas dedicadas al comercio electrónico de ropa y complementos de moda han comprobado que existe un aumento creciente en la demanda de productos mediante vías digitales por parte de los consumidores [1]. Este motivo ha servido como catalizador para aumentar sus esfuerzos en modernizar sus plataformas digitales de venta, buscando simplificar y mejorar así la experiencia del usuario para lograr mayores ingresos.

Si los clientes acuden a un establecimiento físico, un empleado sería capaz de reconocer un artículo que estos le muestren o detallen, con objeto de ofrecerle algo similar dentro del catálogo de productos ofertados. Extrapolado al comercio online, existe un problema para los usuarios, y es la imposibilidad para hacer este mismo tipo de consultas a una persona física. Para solventar este problema las plataformas de venta cuentan con uno de sus componentes más importantes: los motores de búsqueda.

La función de un motor de búsqueda no es otra que la de encontrar productos, dentro un catálogo, con características lo más similares posibles a la información facilitada por el usuario. Multitud de sitios web utilizan únicamente motores tradicionales que funcionan gracias a campos de búsqueda de texto libre y de filtros de selección manual, estos permiten al sistema trabajar directamente sobre los datos estructurados de los productos (talla, género, temporada, color...). Su principal limitación es su incapacidad para trabajar con datos no estructurados, como lo son las propias imágenes de los productos. Es en este punto donde la aplicación de técnicas basadas en Inteligencia Artificial (IA) permitiría aprovechar la información extraída de imágenes facilitadas por el usuario como parte del proceso de búsqueda, para obtener así unas mejores sugerencias, logrando no solo modernizar sus motores de búsqueda actuales sino convertirlos en una importante ventaja competitiva frente a las que puedan disponer otros negocios.

II. Inteligencia Artificial en la App

La aplicación Sugerencias de Moda se presenta como una solución práctica y operativa ante el reto de modernizar los motores de búsqueda clásicos, apoyándose para ello en modelos de Aprendizaje Profundo o *Deep Learning* (DL) característicos del Aprendizaje Automático o *Machine Learning* (ML), una importante rama dentro de la IA. Estos modelos imitan el comportamiento de las redes neuronales humanas, aprenden mediante la observación de sucesos, el ensayo y el error, de ahí el nombre que reciben, Redes Neuronales Artificiales (RNA) [2]. Destacar que los modelos RNA utilizados están fundamentados en uno de los tipos más utilizados para el procesamiento de imágenes, las Redes Neuronales Convolucionales o *Convolutional Neural Networks* (CNN) [3], componente principal de la IA del motor de búsqueda de la aplicación.

Se han generado 4 modelos de CNN distintos pero que comparten una misma arquitectura, esta se divide en más de 9 capas (*layers*) sucesivas e interconectadas de neuronas [4]. Cada una tiene su propia misión y en particular destaca la última, la *softmax*, que genera la salida de la red para la imagen procesada, repartiendo las probabilidades de pertenencia a cada posible valor o clase de la característica a clasificar. Un primer modelo clasifica la imagen según el género del artículo distinguiendo entre masculino y femenino. Para cada género existe su propio modelo que clasifica el tipo de prenda o complemento de moda y el cuarto modelo se encarga de distinguir la clase del color. Cada modelo es sometido primero a una fase de entrenamiento siguiendo durante la misma un proceso de aprendizaje supervisado. Durante esta fase, la red neuronal reajusta de forma óptima los parámetros de cada capa con el fin último de asignar mejor las probabilidades finales. El aprendizaje se realiza utilizando un extenso conjunto heterogéneo de imágenes etiquetadas (*training set*), las cuales han sido procesadas aleatoriamente con distintas alteraciones lumínicas, rotativas, de escala... para poder obtener así unos mejores resultados ante imágenes reales.

Para contrastar el rendimiento obtenido tras dicho entrenamiento, los modelos se someten a una fase de prueba o test, utilizando en esta ocasión imágenes distintas (*test set*) a las del entrenamiento. Por último, son los diversos indicadores estadísticos y medidas de rendimiento los que permiten afirmar que los fines para los que se han diseñado los distintos modelos han alcanzado unos resultados satisfactorios que prueban así la calidad de los mismos y por ende de la IA en la aplicación.

III. Tecnologías y herramientas

El proyecto basa su arquitectura en el modelo cliente-servidor, tal y como se observa en la Figura 1. El cliente, la app, está programado de forma nativa para el sistema operativo Android, utilizando Android Studio [5] como entorno de desarrollo (IDE). Desde el controlador de la app, escrito en código Java, se gestiona la comunicación bidireccional con el servidor comenzando por el envío de peticiones cifradas de extremo a extremo (AES-128) y la posterior recepción de las respuestas obtenidas que se mostrarán a modo de sugerencias.

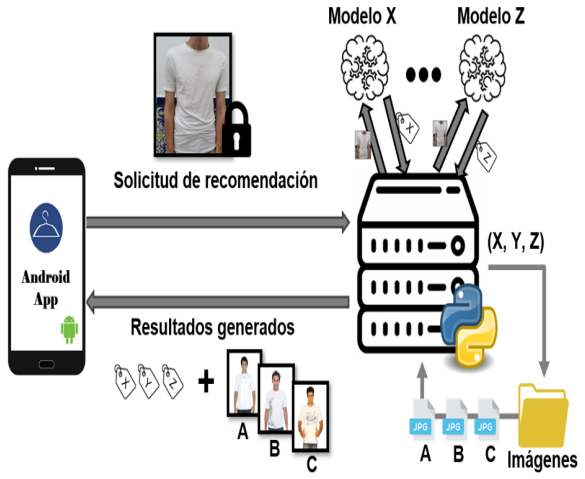


Figura 1. Esquema de la arquitectura cliente-servidor del sistema

El servidor se convierte en la piedra angular del proyecto ya que mantiene operativo el servidor web que gestiona las peticiones de los clientes, haciendo que los modelos CNN ya entrenados procesen cada imagen recibida para generar así los datos necesarios para la lógica de la IA del motor de búsquedas. Con esto se consigue liberar de carga a la app y mantener seguros a los modelos. Tanto el servidor web como los modelos han sido desarrollados en Python, en particular los modelos utilizan integramente el *framework* de desarrollo Keras [6] y asimismo se utilizan diversas librerías para el tratamiento de datos (Numpy), procesamiento de imágenes (PIL), cálculos estadísticos (Scipy)... que permiten completar las fases de aprendizaje y prueba. Toda la aplicación se nutren de un amplio catálogo de imágenes etiquetadas de prendas y complementos de moda [7], el aprendizaje y las pruebas utilizan respectivamente unas 20.000 y 9.000 imágenes distintas.

IV. Utilidad y viabilidad de la aplicación

Debido a la creciente actividad *online* de la sociedad de la información actual realizada mediante dispositivos móviles y tabletas digitales con conexión permanente a internet, cada vez se genera un mayor volumen de datos, en particular datos codificados con formatos de imagen que se almacenan en sus dispositivos personales.

Habrá ocasiones en las que los consumidores quieran y deseen encontrar prendas o accesorios de moda similares al que hayan visualizado en una fotografía, ya sea a través de una red social, de una imagen personal, de una que terceras personas les hayan compartido e incluso de un producto que tengan frente a ellos en ese mismo instante. Sin embargo, es común que las apps de sus proveedores de moda favoritos no puedan obtener este tipo de sugerencias a través de fotografías o que los resultados que les muestren no sean del todo satisfactorios.

Sugerencias de moda es una aplicación capaz de satisfacer esta necesidad, ofreciendo al usuario sugerencias mediante imágenes en tiempo real, de una forma cómoda y segura gracias al proceso de cifrado con la eliminación posterior de la fotografía, con unos tiempos de respuesta generalmente inferiores al de otras alternativas disponibles. Con objeto de darle una utilidad real de negocio y mantenerla como aplicación independiente, las sugerencias podrían redirigir a productos reales pertenecientes a los catálogos de diversas marcas, que el usuario podría finalmente adquirir y adicionalmente durante su uso podría mostrar algún tipo de contenido publicitario monetizado. También sería posible integrar su lógica como una nueva prestación en otras apps de venta de productos de moda ya existentes, utilizando su infraestructura ya preparada para soportar un mayor volumen de usuarios simultáneos.

V. Características de la App

La aplicación desarrollada de forma nativa para el sistema operativo Android, permite a los usuarios que hagan uso de ella obtener sugerencias de moda similares a la prenda o complemento que esté presente en una imagen subida gracias a la IA del motor de búsquedas.



Figura 2. Icono de la aplicación Sugerencias de moda

Para garantizar una buena experiencia de usuario cuando se utiliza la aplicación, el diseño de la interfaz está enfocado en lograr que sea útil, usable y sencilla. La Figura 3 muestra la pantalla inicial o *homepage* que el usuario puede ver una vez que inicia la aplicación. En ella se observa, en este orden, un breve texto explicativo de la aplicación, una introducción visual con dos imágenes que ejemplifican rápidamente cómo se utiliza la app y los dos botones interactivos que permiten al usuario obtener las sugerencias de moda. Por último, una pancarta o *banner* emplazado en la parte inferior para simular contenido gráfico publicitario.

Los usuarios disponen por tanto de dos opciones distintas para obtener las sugerencias a partir de una imagen, ya sea tomando una fotografía directamente con la cámara usando para ello la opción “Utiliza tu cámara” o bien mediante una imagen seleccionada desde la galería del dispositivo a través de la opción “Busca en tu galeria”. Dicha imagen se envía cifrada al servidor web y, una vez procesada, se elimina para garantizar así la privacidad de los datos sensibles del usuario.



Figura 3. Pantalla principal explicativa de la aplicación

El proceso lógico para generar las sugerencias que serán posteriormente mostradas en la aplicación comienza en el lado del servidor, obteniendo en primer lugar las etiquetas o características asociadas al producto que aparezca en la imagen recibida. Para lograrlo, esta imagen se procesa mediante varios modelos de clasificación CNN ya entrenados. En una primera instancia, un primer modelo clasifica el género (femenino o masculino) al que pertenece el producto, y así se distingue cuál es modelo de clasificación del tipo de prenda o complemento que se utilizará en segundo lugar. En el caso del modelo del género femenino se clasifican un total de K_f clases distintas de productos como vestidos, bolsos, pantalones, sandalias... y hasta K_m clases para el género masculino. Por último, se clasifica el color distinguiendo entre K_c clases de colores (negro, blanco, azul, marrón...) reconocidos por su correspondiente modelo. En este trabajo se han utilizado unos valores de $K_f = 20$, $K_m = 16$ y $K_c = 16$.

Con dichas etiquetas ya generadas se escogen n productos (en este trabajo se ha utilizado $n = 3$) que también las compartan de entre los más de 30.000 productos disponibles en el catálogo que compone la base de datos y luego se elimina la imagen recibida por motivos de seguridad. Los productos que resulten escogidos conforman las sugerencias de moda y sus imágenes junto a las propias etiquetas se envían a la aplicación.

Tras esto, la app se encarga de procesar los datos recibidos para mostrarlos a continuación en una última pantalla al usuario. Destacando, en primer lugar, las características que se han podido reconocer de la prenda o accesorio (género, color y categoría o tipo) que aparezca en la imagen y por último las imágenes de los productos que conforman las sugerencias de moda obtenidas para la consulta realizada (ver Figura 4).



Figura 4. Proceso de obtención de una sugerencia para una imagen tomada desde la cámara del dispositivo móvil

VI. Conclusiones

Las datos no estructurados como lo son las imágenes están cada día más presentes en los dispositivos tecnológicos de la sociedad de la información actual, gracias a la aplicación de técnicas de IA la app desarrollada ha podido demostrar cómo

es posible sacarles partido a estos datos de una forma comoda, práctica y segura, rompiendo así con el estigma de utilizar siempre las tradicionales búsquedas por texto para buscar lo que necesitemos o estemos interesados en encontrar.

Con objeto de ampliar este proyecto para ofrecer sugerencias más precisas, cabría destacar la posibilidad de desarrollar nuevos modelos. Destacando un modelo que realice reconocimiento múltiple de objetos, permitiendo ofrecer sugerencias también para aquellos casos en los que aparezcan varios productos a la vez en una misma imagen. También se pueden desarrollar otros nuevos métodos de clasificación que permitan generar nuevas etiquetas sobre otras características de los productos como pueden ser la estacionalidad, el estampado o la textura, entre otras.

ANEXO

Se ha creado a un breve video[1] sobre la aplicación. A lo largo del mismo se resumen y explican sus principales funcionalidades, así como las técnicas de IA en las que se apoya. En cuanto a la propia aplicación, esta se encuentra disponible para su descarga [2].

AGRADECIMIENTOS

Este trabajo ha sido financiado parcialmente por los proyectos RTI2018-094645-B-I00 y UMA18-FEDERJA-084. Todos los proyectos incluyen fondos FEDER. Los autores quieren agradecer los recursos computacionales y asistencia técnica proporcionados por el Servicio de Supercomputación y Bioinformática (SCBI) de la Universidad de Málaga y por el Instituto de Investigación Biomédica de Málaga – IBIMA. Los autores también quieren agradecer la ayuda de la Universidad de Málaga.

REFERENCIAS

[1] (2021) Informe 2020: El sector de la moda española online (actualizado abril 2021). [Online]. Available: https://www.salesupply.es/media/salesupply-blog/2021/05/19/informe-2019-el-sector-de-la-moda-espa%C3%B1ola-online/

[2] F. L. R. F.-F. J. Lévy Mangin, J.P., *Las redes neuronales artificiales*. NetBiblo, 2008.

[3] Ujjwalkarn. (2016) An intuitive explanation of convolutional neural networks. [Online]. Available: https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets

[4] M. J. V. Hilera R., J, *Redes neuronales artificiales. Fundamentos, modelos y aplicaciones*. RA-MA, 1995.

[5] Google, "Android studio." [Online]. Available: https://developer.android.com/studio#downloads

[6] F. Chollet *et al.*, "Keras," https://github.com/fchollet/keras, 2015.

[7] P. Aggarwal, "Fashion product images dataset," 2019, data retrieved from Kaggle, https://www.kaggle.com/paramaggarwal/fashion-product-images-dataset.

[1] https://drive.google.com/file/d/14CCdiXtJLFQtS6-VPjC-MHtBUOEzakNG

[2] https://drive.google.com/drive/folders/1tS8WxJtkj5iAaISGZgWXwvBDGDllaY5_

Competición CAEPIA-App: NBA Analyzer, una app para el apoyo a entrenadores analizando datos de la NBA

Jose Luis Ruiz-Casado, Rafael Marcos Luque-Baena, Miguel A. Molina-Cabello
Departmento de Lenguajes y Ciencias de la Computación. Universidad de Málaga, España
Instituto de Investigación Biomédica de Málaga – IBIMA, España
Emails: joseruizcasado02@gmail.com; {rmluque,miguelangel}@lcc.uma.es

***Resumen*—NBA Analyzer trata de solventar un problema que afecta a los entrenadores de la liga de baloncesto norteamericana: el poco tiempo de preparación para un partido de liga regular. Debido a la gran cantidad de partidos que se disputan a lo largo de la temporada regular, es muy común que los equipos jueguen varios partidos en una semana; incluso en un intervalo de tres días se suelen jugar dos partidos. Para ello en este trabajo se propone una aplicación que analiza los datos que la NBA pone a disposición de los equipos en forma de ficheros tras la disputa de cada partido. Estos datos recogen todos los eventos sucedidos durante el partido, y con ellos la aplicación desarrollada calcula estadísticas tanto avanzadas como simples de cada jugador. Estas estadísticas se utilizan para alimentar dos modelos de inteligencia artificial: uno para la predicción de partidos y otro para la agrupación de jugadores por perfil. Además, gracias a la información sobre el posicionamiento de cada jugador en cada evento registrado y a dicho agrupamiento, la aplicación permite ofrecer a los entrenadores información sobre qué tipo de jugador de su equipo tiene mayor eficacia defensora sobre cada jugador del equipo contrario.**

I. Introducción

La Asociación Nacional de Baloncesto [1], en adelante NBA (*National Basketball Association*), es la liga privada de baloncesto profesional de los Estados Unidos que fue fundada en 1946. Al ser una liga gestionada por una entidad privada las plazas son elegidas por la propia NBA de manera que se asignan en función de ciudades de Norte América y una entidad o grupo financiero crea o compra una marca, a esto se le conoce como franquicia. En la actualidad la liga está formada por treinta franquicias, y se dividen en dos conferencias, la este y la oeste, las cuales se encuentran a su vez divididas en tres divisiones de cinco equipos cada una. Estas divisiones se realizan por la localización geográfica de la ciudad que representa cada franquicia.

A día de hoy la NBA es considerada como la mejor liga de baloncesto del mundo, y una de las ligas deportivas con mayor impacto. Esto se debe principalmente a dos factores: la calidad de los jugadores y de los equipos es muy superior a la del resto y la inversión económica que recibe la liga y los equipos. En 2014 se firmó un nuevo contrato televisivo con las empresas *ESPN* y *TNT* por nueve años de veinticuatro billones de dólares, lo que supuso un gran crecimiento en la capacidad financiera de los equipos. Esta inversión repercute de manera directa en las franquicias, ya que se llevan parte de ella debido a los derechos televisivos que poseen sobre los partidos que juegan. Tanto repercutió que ese mismo año los estadios de cada franquicia se equiparon con sistemas software para el seguimiento de los jugadores durante los partidos mediante cámaras. Esto supuso un importante cambio de paradigma en el análisis del baloncesto. El sistema de seguimiento permite obtener las coordenadas de los diez jugadores sobre la cancha y las coordenadas y velocidad de la pelota, de manera que permite un estudio más exhaustivo del impacto de los jugadores sobre el juego. Esto también implicó que las franquicias NBA incorporasen un departamento propio dedicado al análisis de datos.

II. Descripción de la aplicación

Con esta aplicación desarrollada se pretende agilizar la labor de los entrenadores encargados de analizar a los rivales antes de cada partido. Para ello se construirán estadísticas simples y avanzadas de cada jugador y cada equipo en función de los datos de cada partido proporcionado por la propia NBA.

Para calcular las estadísticas simples se utilizan los datos de eventos de cada partido que proporciona la NBA. Un evento es una acción que sucede en un partido de la NBA, por ejemplo un intento de canasta, un robo o una falta. Estos datos son analizados para poder calcular las estadísticas simples a partir de los eventos del partido. Una vez se hayan calculado la estadísticas simples de los jugadores y equipos que participan en el partido, se realizan los cálculos para obtener las estadísticas avanzadas. Los valores calculados son añadidos a la información de cada jugador y cada equipo y almacenados en una base de datos. En la Figura 1 se observa cómo se muestran estos datos en la aplicación, de manera que la información más relevante puede ser visualizada de manera rápida gracias a diferentes gráficas.

Con los datos de posicionamiento que ofrece la NBA se extrae información posicional sobre cada tiro realizado por un jugador, para poder calcular el mapa de calor de un jugador. Esta información también se utiliza para extraer qué jugador defiende al tirador y se crea un conjunto de datos con información defensiva, de manera que se puede obtener el número de defensas exitosas realizas por cada defensor. Con esta información se pueden generar gráficas en las que se le

ofrece al entrenador la densidad de tiros desde los distintos puntos de la cancha, como se observa en la Figura 2.

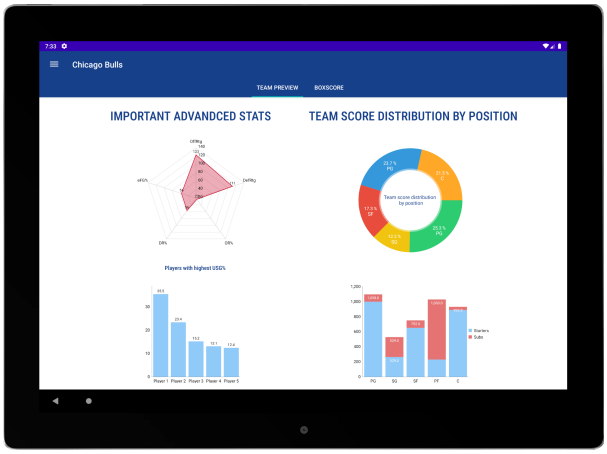


Figura 1. Visualización de estadísticas.

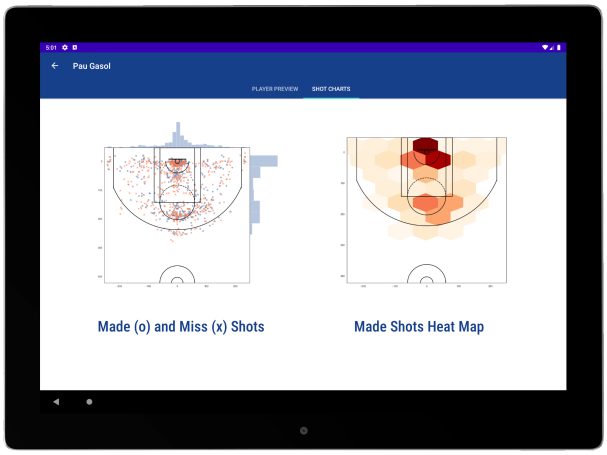


Figura 2. Visualización de estadísticas.

Mediante un modelo de agrupación se crean para cada posición tradicional del baloncesto (base, escolta, alero, alapívot y pívot) grupos de jugadores con un estilo de juego similar. Esta información unida a la información defensiva hace que la aplicación sea capaz de identificar a qué tipo de jugador defiende mejor cada jugador. Para facilitar la labor del entrenador, la aplicación muestra una alineación recomendada con los jugadores de su equipo que mejor defienden al quinteto del equipo rival.

Además, la aplicación es capaz de realizar un informe del hipotético partido frente a otro equipo, además de la predicción de dicho partido. En este informe se enfrentan las estadísticas avanzadas de los equipos que disputarían el partido para facilitar la comparación de aspectos claves que pueden decantar el enfrentamiento.

Las funcionalidades que incluye la aplicación son las siguientes:

- Cálculo de estadísticas simples y avanzadas tanto de jugadores como de equipos.
- Análisis de partidos mediante los archivos que proporciona la NBA.
- Realización de informes de enfrentamientos entre dos equipos.
- Predicción de partidos.
- Recomendación de mejor quinteto defensivo contra un equipo rival.
- Visualización de la información almacenada en la base de datos.

Las funcionalidades de la aplicación suponen una comunicación en tiempo real con el microservicio adecuado, para asegurar así que la información que se muestra está actualizada.

III. Arquitectura de la aplicación

La aplicación consta de dos partes claramente identificables: el servidor y la aplicación móvil. El servidor se encarga de realizar todos los cálculos estadísticos que requiere la aplicación. También alberga los modelos de inteligencia artificial que se consultan desde la aplicación. La arquitectura que se utiliza para definir el servidor es una arquitectura basada en microservicios, que permite aislar las distintas funcionalidades. Para que la aplicación pueda comunicarse con los microservicios se define una API REST, de manera que la aplicación se comunica con la API y ésta reenvía la petición al microservicio adecuado, que responde con la información adecuada y es transmitida a la aplicación.

Los microservicios definidos para la aplicación son:

- Cálculo de estadísticas. Mediante la información procesada de los partidos este subsistema calcula las estadísticas para los jugadores y equipos implicados en el partido.
- Análisis de partidos. Este microservicio se encarga de analizar los ficheros con los datos de los partidos.
- Gestor de base de datos. Implementa las operaciones necesarias para almacenar datos en la base de datos y realizar las consultas oportunas.
- Inteligencia Artificial. Este microservicio será el encargado de almacenar y gestionar los modelos de inteligencia artificial que se utilizan.

III-A. Tecnologías utilizadas

Para el desarrollo de la API REST se ha utilizado el entorno de trabajo Django. El desarrollo en Django es muy rápido debido a que incorpora múltiples herramientas para facilitar tareas tediosas del desarrollo. También incorpora muchos paquetes con funcionalidades ya desarrolladas haciendo posible una rápida implantación, ya que tan solo sería necesario adaptarla. Otro punto bastante importante para la selección de este entorno es la interfaz para acceso a la base de datos, que es muy completa y facilita la realización de consultas.

Para el desarrollo de los modelos de inteligencia artificial se ha utilizado Scikit-Learn. Esta librería incluye multitud de funciones y algoritmos para el aprendizaje automático y permite la implantación de un modelo de aprendizaje automático

de manera sencilla. Para el manejo de los datos se utiliza Pandas, que organiza la información en DataFrames. Estos objetos son similares a tablas y disponen de una gran cantidad de operaciones optimizadas.

La aplicación cliente que se ha desarrollado para dispositivos Android se ha utilizado el lenguaje de programación Kotlin, que es un lenguaje de tipado estático ejecutado sobre una máquina virtual Java y que actualmente cuenta con soporte oficial de Google, además del apoyo de los desarrolladores del propio lenguaje, que son JetBrains.

IV. Inteligencia Artificial en NBA Analyzer

En la aplicación se utilizan dos modelos de Inteligencia Artificial que responden a dos tipos de problemas distintos. Los problemas a los que se enfrenta la aplicación son: **Clasificación binaria** y **Agrupación no supervisada**. La clasificación binaria permite separar el conjunto de datos en dos clases opuestas, de manera que podemos distinguir en el conjunto de datos los registros que pertenecen a la clase positiva y los que pertenecen a la clase negativa. Por su parte, la agrupación no supervisada categoriza la información para generar segmentaciones en los datos, de manera que quedan agrupados los datos que tienen características similares.

En el problema planteado en el presente trabajo, la clasificación binaria se presenta a la hora de tratar de predecir qué equipo ganará un partido, donde lo que se pretende clasificar es si el equipo local va a ganar o no el partido. En cuanto a la agrupación no supervisada, trata de agrupar a los jugadores en conjuntos de manera que los que sean de un mismo grupo tengan estadísticas similares. Para ambos casos se utilizan las estadísticas avanzadas como datos de entrada. Estas estadísticas fueron definidas por Dean Oliver en su libro *Basketball on Paper* [2] y se utilizan para valorar un gran número de aspectos importantes del baloncesto, tales como conocer el desempeño ofensivo y defensivo de un equipo más allá de los puntos anotados y concedidos.

IV-A. Predicción de partidos

La predicción de los partidos se hace en función de las estadísticas avanzadas de los equipos que participan en el partido. Para afrontar este problema se han evaluado distintos modelos para tratar de encontrar el más adecuado. Los modelos que han sido evaluados son: *Logistic Regression* [3], *Support Vector Machines* [4], *Random Forest* [5] y *Neuronal Network* [6].

Estos modelos fueron creados mediante el paquete scikit-learn [7] y evaluados utilizando la técnica **K-fold** [8]. Esta técnica consiste en que los datos son divididos en K subconjuntos, de manera que uno de los subconjuntos es utilizado como datos de prueba y el resto $(K-1)$ como datos de entrenamiento. Este proceso se repite durante K iteraciones, con cada uno de los posibles subconjuntos. El rendimiento de cada modelo durante cada iteración es almacenado y, una vez que se han realizado todas las iteraciones, se calcula el rendimiento medio de cada modelo. Para ello se utilizan medidas como la precisión, desviación típica o las matrices de confusión. Estas medidas ayudan a entender qué modelo tiene mayor capacidad de generalización. Tras estos experimentos, se obtuvo que el modelo que presentaba mejor rendimiento fue el Random Forest.

Las predicciones de este modelo se realizan sobre los datos que se almacenan en la base de datos. Cuando un usuario selecciona un equipo en la interfaz se realiza una petición a la base datos. Las estadísticas avanzadas no son almacenadas en la base de datos y son calculadas en el momento, de forma que permite tener una base de datos ligera. Antes de realizar la respuesta se hace la predicción usando las estadísticas del equipo seleccionado por el usuario y por el equipo que el usuario seleccionó en el momento de registrarse en la aplicación. En la Figura 3 se puede observar cómo se muestra la predicción, que se acompaña de las estadísticas avanzadas que se utilizaron para la predicción.

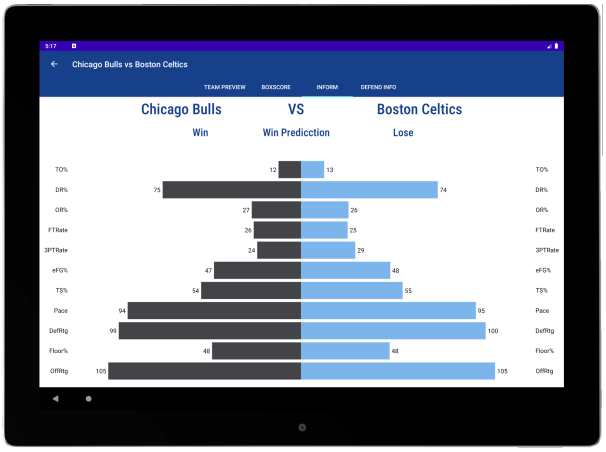


Figura 3. Informe de enfrentamiento.

IV-B. Agrupación de jugadores

Dado que se pretenden encontrar grupos de jugadores con un estilo de juego similar (es decir, un perfil) para así poder obtener información valiosa, para cada una de las cinco distintas posiciones del baloncesto se crea un modelo distinto de agrupación. El algoritmo que se utiliza para estos modelos es **K-Means** [9]. Este método agrupa los datos en K grupos diferentes, de manera que cada dato pertenece únicamente a un grupo. Dado que es necesario indicar el número de grupos que se van a formar (K), para tratar de encontrar un valor adecuado para este parámetro se ha utilizado el método conocido como **Elbow Curve**. Este método se basa en mostrar en una gráfica el valor de inertia del modelo en función del número de clústers, siendo la inertia una métrica que mide la distancia entre los puntos que forman cada grupo (o clúster). El objetivo es tratar de minimizar el valor de la inertia pero manteniendo un número adecuado de grupos, de forma que en la gráfica se puede analizar visualmente qué valor de K es adecuado.

Con estos grupos por posición se pretende crear información defensiva sobre los jugadores. Para ello se analizan los datos de posición y para cada tiro realizado en cada partido se extrae el

tirador y el defensor y si el tiro ha sido anotado o no. También se añade la información sobre el grupo y posición del defensor y del tirador. Con esta información se puede obtener a qué grupo defiende mejor cada jugador y cada grupo en general, ofreciendo una información de apoyo útil al entrenador. En la Figura 4 se muestra la recomendación que hace la aplicación sobre qué jugador de nuestro equipo sería el mejor defensor para cada jugador del quinteto del equipo contrario.

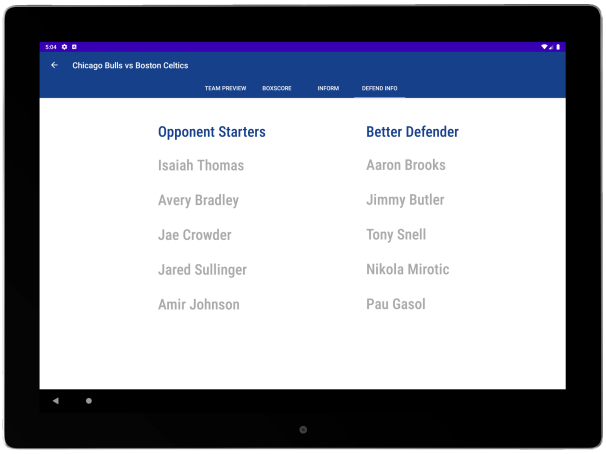


Figura 4. Selección de mejor quinteto defensivo.

V. Utilidad y viabilidad de la Aplicación

La NBA ha realizado una apuesta clara por la tecnología y los datos para mejorar la calidad de la propia liga. Esto ha constituido un ecosistema que permite que surjan aplicaciones que utilizan tecnología puntera para crear datos de calidad respecto a cada partido. La aplicación propuesta en este trabajo (NBA Analyzer) trata de facilitar la labor que realizan los analistas, apoyándose en técnicas de inteligencia artificial.

En la NBA, el equipo de analistas se basa en aplicaciones que necesitan la introducción manual de los datos que, teniendo en cuenta el poco tiempo entre partidos, supone una ardua tarea. NBA Analyzer automatiza el proceso de recolección de datos, agilizando así el proceso de análisis. Adicionalmente se realiza un procesado de estos datos, tanto con técnicas de análisis estadísticos e Inteligencia Artificial. De manera que al analista o entrenador que utilice la aplicación se le muestra información útil, facilitando el entendimiento de los datos.

Por su parte, la adaptación de NBA Analyzer a otras ligas solo requiere de la modificación del servidor, de manera que se adapte al formato de datos que se tenga para dicha competición. Poder adoptar la aplicación a entornos más amateurs ayudaría a que la formación de entrenadores y analistas mejore, ayudando a focalizar en los aspectos clave para entender mejor a los equipos rivales.

VI. Conclusiones

Mediante la utilización de los modelos de inteligencia artificial y las técnicas de análisis estadístico se ha conseguido mostrar información avanzada sobre los distintos equipos y jugadores de la NBA. El objetivo de condensar la información procesada en un formato visual e informativo a partes igual se ha conseguido con resultados notables mediante la generación de diferentes gráficas que hacen entender de manera rápida y ordenada aquella información más relevante para el entrenador.

La inteligencia artificial juega un papel crucial para la aplicación ya que permite generalizar los datos de manera que la aplicación tiene la capacidad de brindar información útil, incluso cuando los datos recogidos para un equipo o jugador concreto son escasos.

Como trabajo futuro, el objetivo a largo plazo de esta aplicación es poder adaptarla a la liga de baloncesto nacional. Para lograr este propósito habría que realizar leves modificaciones en la forma en la que la aplicación adquiere los datos. En cuanto a los modelos de inteligencia artificial, estos deberían ser reentrenados ya que el estilo de juego de la liga nacional es bastante diferente al de la NBA. Sin embargo, los pasos de este proceso serían los mismos que los aplicados en este trabajo.

ANEXO

Una muestra de las funcionalidades de la aplicación desarrollada puede verse en el siguiente video[1], mientras que los códigos de la app (cliente)[2] y del servidor[3] son públicos.

Agradecimientos

Este trabajo ha sido financiado parcialmente por los proyectos RTI2018-094645-B-I00 y UMA18-FEDERJA-084. Todos los proyectos incluyen fondos FEDER. Los autores quieren agradecer los recursos computacionales y asistencia técnica proporcionados por el Servicio de Supercomputación y Bioinformática (SCBI) de la Universidad de Málaga y por el Instituto de Investigación Biomédica de Málaga – IBIMA. Los autores también quieren agradecer la ayuda de la Universidad de Málaga.

Referencias

[1] NBA. National baskteball association - wikipedia. [Online]. Available: https://es.wikipedia.org/wiki/National_Basketball_Association
[2] D. Oliver, *Basketball on Paper*, 2004.
[3] D. W. Hosmer and S. Lemeshow, *Applied logistic regression*. Wiley New York, 2000.
[4] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.
[5] T. K. Ho, "Random decision forests," in *Proceedings of 3rd international conference on document analysis and recognition*, vol. 1. IEEE, 1995, pp. 278–282.
[6] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *The bulletin of mathematical biophysics*, vol. 5, no. 4, pp. 115–133, 1943.
[7] Scikit-Learn. Scikit-learn. [Online]. Available: https://scikit-learn.org/stable/
[8] M. Stone, "Cross-validatory choice and assessment of statistical predictions," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 36, no. 2, pp. 111–133, 1974.
[9] J. MacQueen *et al.*, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, vol. 1, no. 14. Oakland, CA, USA, 1967, pp. 281–297.

[1] https://drive.google.com/file/d/1qIjPwhC7t5yVkToJgPsnq1xiFZvl_WO3
[2] https://drive.google.com/file/d/1i_o23SdHaVSM6CMb359hGDyRnDaUDzKx
[3] https://drive.google.com/file/d/1fFvY3oLdM30RRQy2o7kx-DTRdFPuO9xy

CAEPIA-App Competition: Sharpmony: A Computational Intelligence based tool for 4-part harmony

Francisco Fernandez de Vega
Departamento de Tecnología de los Computadores y de las Comunicaciones
Universidad de Extremadura
Mérida, España
fcofdez@unex.es

Jorge Alvarado Díaz
Departamento de Tecnología de los Computadores y de las Comunicaciones
Universidad de Extremadura
Mérida, España
jorgealvaradodiaz@gmail.com

Martin Morita-Hernández
Departamento de Tecnología de los Computadores y de las Comunicaciones
Universidad de Extremadura
Mérida, España
martin.morita.13@gmail.com

***Abstract*—Sharpmony [1] is the first AI-based APP that provides a set of tools for music harmony students and teachers. On the one hand, a kernel has been developed with the assistance of experts in the area, that encodes classic harmony rules. This kernel allows to automatically detect errors in SATB scores that students develop when learning four-part harmony in music conservatories. Secondly, the kernel is part of the fitness function that an evolutionary algorithm uses to automatically provide solutions to this kind of composition exercises. Finally, a Deep Learning approach is used to capture handwritten SATB scores, so that students can take a photograph of their composition exercise, and then automatically enter the review process within Sharpmony, that will finally offer a PDF with errors detected.**

After the tool was published in Google Play in 2019, 12 institutions and more than 1800 students are using Sharpmony, with several million chords, notes and errors checked.

***Index Terms*—Evolutionary Algorithms, Deep Learning, Expert System, Music Composition.**

I. Introduction

In classical music, the study of harmony is one of the key elements for students to develop their music composition skills. Western music has developed and evolved a set of harmonic rules that are learned and applied by all students enrolled in professional music conservatories. These rules, which began their development, more or less formally, in the *Renaissance*, were definitively established in the *Baroque*, with the study of counterpoint and fugue, Bach being the main composer of the period. In addition to instrumental music, Bach developed a whole series of choral compositions, in which the overlapping of voices require skill and mastery in harmonic development.

Thus, in the period that followed, *classicism*, the study of four-voice choral harmony -composed of four voices: Soprano, Alto, Tenor and Bass (SATB), is consolidated; and new rules emerge that establish the novelty of the music of this period.

[1]For Downloading Sharpmony, search it at Google Play, or directly go to https://n9.cl/fuv7a and link video https://youtu.be/ThSV8qMILIQ

It is therefore classicism's set of rules the one that is currently studied in the subjects Harmony 1 and 2. Students must thus compose SATB chorales in their third and fourth year of professional music education. The study of harmony thus consists of taking a melody as the starting point, melody that is assigned to the Soprano, and the student must compose the rest of the voices following rules that state what is allowed and not allowed. Similarly, students can begin with a figured Bass and try to develop the other voices.

The rules affect both the intervals between voices and the melodic movements that each voice develops in combination with the rest of the voices, as well as chord progressions.

To the best of our knowledge, and although the problem of SATB composition has been previously addressed in the Artificial Intelligence literature, there have been no attempt to develop a useful APP that allows both students and instructors to improve the learning process by means of Artificial Intelligence techniques.

II. Brief description of the main functionality of the App

Sharpmony is therefore the first APP that provides a series of tools that make it easier to harmony students to improve their learning curve. On the other hand, teachers are provided with an improved APP specially designed for instructors, together with a cloud infrastructure that allows them to assign tasks, authomatically check exercises and keep a record of their students improvements. Namely, Sharpmony provides:

- A Deep-learning based tool for capturing and analyzing hand-writen SATB scores.
- A simplified score editor for SATB harmony.
- An AI-kernel based tool to review students exercises.
- A series of tools that allow teachers to assign exercises to students and check their progress.
- An Evolutionary Algorithm that allows teachers to evolve solutions to problems.

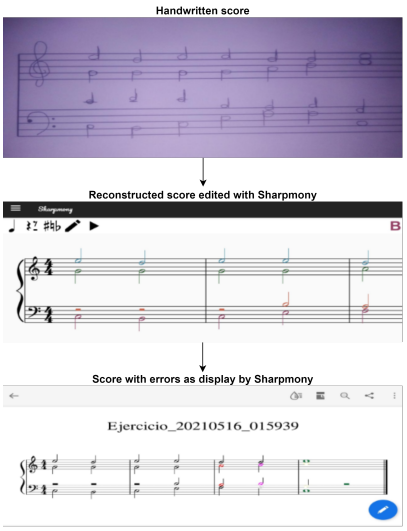


Fig. 1. Student exercise and its authomatic review provided by Sharpmony

In order to better understand what Sharpmony offers, we show in figure 1 an example of choral harmony exercise developed by a student, what sharpmony produce once Optical Music Recognition is applied, and finally, harmony errors detected by the kernel.

As we notice, four different voices are present in SATB exercise, which produce a chord progression. The main goal for students is to be able to compose four-part harmony exercises with no errors.

Thus, taking into account the way students work, sharpmony provides:

- An Optical Music Recognition method to enter handwritten SATB scores, as well as an score editor.
- The tools allowing interaction with teachers: Managment of teachers assignments, edition of scores to solve the exercises, etc.
- Connection with the kernel in charge of checking exercises.

All these tasks requiere not only the APP (what the student uses), but also a cloud infrastruture that centralize the management of institutions (conservatories), users, exercises, etc: the backend in the cloud required for everything to properly work.

On the other hand, considering teachers, Sharpmony provides:

- A method to assign exercises to students or group of students.
- The required tools to check exercises and the authomatic reviews provided by Sharpmony.
- We also provide AI based composition methods that authomatically solve SATB problems.

We decided to develop an APP given the kind of devices that usually young students use: hand-held devices. Moreover, students typically use android based devices, so we first developed an Android version of Sharpmony, although we

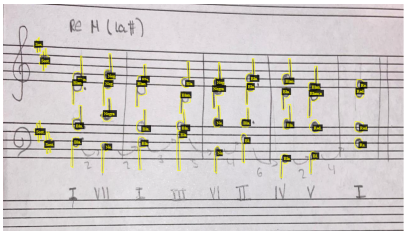


Fig. 2. Object segmentation for the learning process

hope to have the iOS version ready for the next accademic year. Yet, a series of additional tools are available on the project website [2], so that instructors can perform managment operations, such as deciding the kind of error controls to be checked on exercises, create and manage group of students, etc.

III. I.A. Technique(s) on which it is based: Computational Intelligence applied to 4-part harmony learning

A series of IA techniques have been applied to develop every tool included within Sharpmony. Particularly, i) a Deep Learning approch has been applied to the OMR tool; ii) a Kernel with harmonic rules have been developed, thus providing an expert system developed using functional programming, that is then iii) embodied within a fitness function of an Evolutionary Algorithm. We include below a summary of the algorithms used as well as references to papers that describe them with more detail.

A. Deep Learning applied to SATB handwritten scores

Four general stages have been applied to build the Optical Music Recognition System that Sharpmony provides:

- First, we proceeded to create our repository consisting of photographs of the handwritten scores, to which the segmentation process is applied.
- These already segmented scores have then been used to train a pre-trained deep convolutional neural network, and finally a classification of the various symbols contained in each image of the repository is obtained. This procedure is performed in a first training stage for our problem.
- Then, the testing stage is performed on unknown images for the network.
- Finally, once the trained network is ready, a server is set up with a python version of the trained network, in charge of receiving photographs taken by Sharpmony APP, and converting them to an MusicXML file.

Figure 2 shows a segmented handwritten score used during the training step.

On the other hand, figure 3 shows a sample of detected objects in a score. Although the quality of results in the current implementation is high, around 90%, errors produced must be fixed before the score is analyzed. This will be necessary

[2]https://sharpmony.unex.es

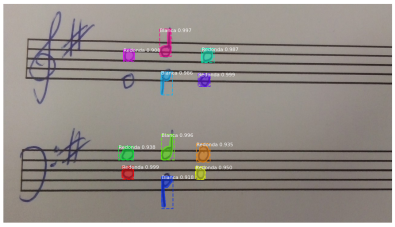


Fig. 3. Object recognition result

until 100% is reached in the optical music recognition process. Although we are optimistic about the results we may reach in the comming months, we have developed and included within Sharpmony a frendly score editor, that allows students to check the MusicXML produced, as well as directly edit a new score from scratch (figure 1 shows the score editor in the center).

Figure 4 shows the architecture of the OMR layer within Sharpmony. Detailed information about the Mask-R-CNN model applied in this stage is provided in [1], whose final result is an MusicXML standard file.

B. Classical harmony rules kernel

Once the MusicXML file is ready, the AI kernel in charge of checking harmonic rules is applied to the exercise. A server is in charge of receiving MusicXML files, checking all rules previously selected by the teacher, and providing a PDF version of the score with wrong notes and chords marked with different colors associated with rules broken.

Figure 1 shows the score reviewed, with some notes marked with colors, corresponding to notes that do not comply with the rules.

Specifically, errors corresponding to *Distances greater than octave between voices (light yellow), Incorrect Cipher (mustard yellow), Parallel Octave-Fifths (red) and incorrect chords (olive green)* are marked. All the rules have compiled by our experts team, music instructors working in Professional Music Conservatories in Extremadura. More information about the error controls is available in the website.

C. Evolving 4-part harmony scores

As described above, we developed a kernel with the harmonic rules we had selected. This kind of expert system is then embodied within a fitness function that is used by a Genetic Algorithm to provide solutions to any melody we may write in Sharpmony.

We decided not to provide this powerful tool within the standard version of Sharpmony because of two main reasons: i) On the one hand we want students to develop their hability to compose SATB scores, and providing a tool that solve the problems is not desirable; ii) Because of the difficulty of the problem, hours or days are required to find solutions of quality.

The evolutionary algorithm employed is based on the results already published in [2], which interested readers may check for every detail of the algorithm configuration. Basically, the algorithm works in a two-stages approach:

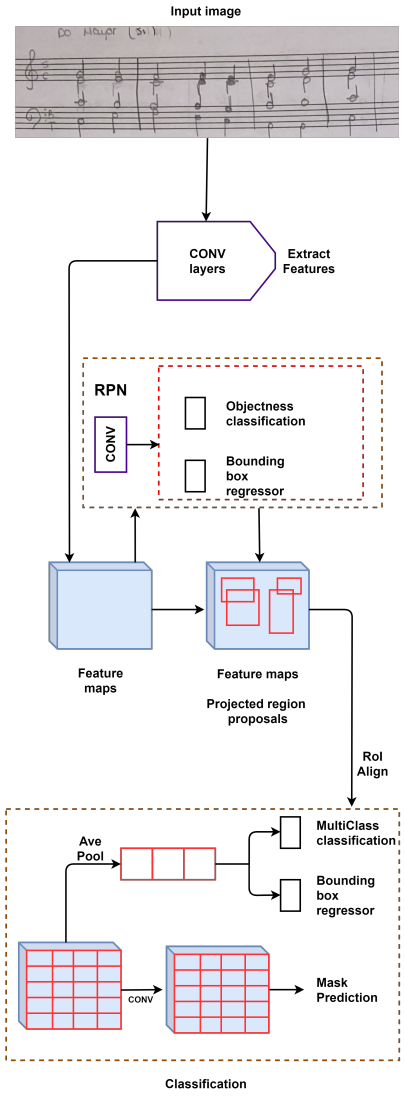


Fig. 4. Mask-R-CNN

- First, the algorithm tries to find a progression of chords that fits the melody provided. This means that for each of the notes in the melody, a suitable chord must be found, and the series of chords must be *reasonable*, (such as II-V-I progression).
- Second, once the progression is available, a distribution of chord notes must be found so that the number of rules broken are minimized in the score. The kernel within the fitness function is in charge of evaluating results.

The evolutionary algorithm, as well as the Deep Learning based algorithm, are run on a server hosted by the University of Extremadura. Given that computing resources are limited, and the large number of users registered (more than 1800) we cannot provide yet the tool to all of them. Instead, we are working with teachers to polish the tool, analyse solutions provided by the evolutionary algorithm, and find new ways to

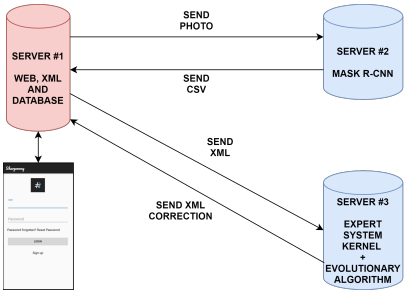


Fig. 5. Sharpmony Cloud Infrastructure

reduce computing time.

IV. Technologies and tools used

The development of the application has been natively programmed from scratch, for Android (already available in Google Play, Spanish and English versions) and iOS (under development) devices using the proprietary technologies offered by each platform, offering a unique user experience and a lightweight application size so that it can be run on a wide variety of mobile devices. MusicXML has been used as an open standard for XML-based music notation, thus offering compatibility with a wide variety of music score editing tools (such as Finale, Sibelius, Musescore, Noteflight, etc.).

On the server side, the back-end system is hosted by the University of Extremadura. Three different servers are in charge of each of the componentes required to support Sharpmony tasks, as depicted in figure 5.

The genetic algorithm as well as the Kernel (Expert System with harmonic rules, that is also part of the fitness function) has been developed using the Common LISP programming language. There are several reasons behind choosing functional and symbolic programming as the framework (check for instance [3]), mainly due to the highly symbolic and redundant nature of music representation within scores. Although music is made up of different sounds that are mixed together, and every sound can be analyzed by means of its frequency, the way music is represented is highly redundant and symbolic (for instance, a frequency of 440Hz can be described as notes A, Gx or Bbb).

For the development of both the backend and the frontend of the web platform, the PHP-based Yii 2.0 framework has been used.

Finally, for client-server communication, a REST web service has been developed, which is consumed by the application through the HTTPS protocol for secure data transfers.

V. Usefulness and feasibility of the App

The mobile application is one of the main factors behind the use of Sharpmony by 12 institutions and 1800 students. Although the main functionality of the kernel had been developed several years ago, and we began was tests with some students by means of a file sharing service in 2015, we could not reach a wide audience until the Android APP was completed by 2018. Moreover, although the score editor and the kernel in charge of scores reviews were available a year ago, we have only very recently added the Computational Intelligence based tools described above, that allow to both evolve a composition (only for teachers), and capture and convert handwritten scores into MusicXML standard files, that allows further editing and correction, if necessary, in the app itself, or thanks to the compatibility offered, in a third party tool.

The music notation editor has been specifically designed to be used by both conservatory students and users with no previous experience in music notation software. Unlike the tools available for music editing, with a fairly expensive learning curve, the APP developed is specially adapted to beginner users, and to Harmony and Composition subjects of professional conservatories, removing much of the usual complexity of the editors, and keeping only the elements necessary for this type of exercises.

Two types of benefits associated with each of the application's user profiles can be highlighted:

- In the case of students, they can send exercises to be reviewed, either from a photograph or using the editor, as many times as necessary, without waiting for their teacher to correct the exercises in class. Thus, the system allows a review history, and the student can apply the correction/improvement cycle as many times as he/she wishes until an error-free exercise is achieved.
- Teachers can propose exercises in the APP, associate them to groups of students previously configured, select harmony rules to be applied during the authomatic review process, and finally check the exercises developed by the students. In addition, they can use the tools to automatically solve through evolutionary procedures, and use the results in more advanced conservatory subjects: such as composition fundamentals, choir, etc.

We plan to launch the iOS version of Sharpmony in September 2021, and hope to reach a much wider audience, thus allowing Sharpmony to be the standard tool for learning 4-part harmony.

Acknowledgment

We acknowledge support from Spanish Ministry of Economy and Competitiveness under project TIN2017-85727-C4-{2,4}-P and RTI2018-095180-B-I00, Regional Government of Extremadura, Department of Commerce and Economy, the European Regional Development Fund, a way to build Europe, under the project IB16035 and Junta de Extremadura, project GR18049.

References

[1] M. Morita, J. Villegas, F. Fernández de Vega, "Aplicación de técnicas de Deep Learning alreconocimiento optico de partituras SATB" Proceedings MAEB 2021. To be published.

[2] F. Fernández de Vega, "Revisiting the 4-part harmonization problem with GAs: A critical review and proposals for improving." Proceedings IEEE CEC 2017: 1271-1278

[3] F. Fernández de Vega: To Be, or Not To Be: That is the Recursive Question. IEEE EDUCON 2019: 1294-1299.

CAEPIA-APP Competition: BIN-CT

Juan José Guerrero*, Javier Ferrer†, Rubén Saborido†, Enrique Alba†
*Lenguajes y Ciencias de la Computación, Universidad de Málaga, España
†ITIS Software, Universidad de Málaga, España
Email: jjguerrero@lcc.uma.es, ferrer@lcc.uma.es, rsain@uma.es, eat@lcc.uma.es

I. Introducción

En los últimos tiempos, nuestro planeta ha experimentado tanto un crecimiento demográfico significativo como una revolución tecnológica sin precedentes que se traducen en una demanda de producción cada vez mayor. Esta creciente demanda ha causado que la generación de residuos y el control de la polución se conviertan en grandes problemas que, actualmente, no son atendidos en la medida que se necesita [1], [2]. Estos problemas tienen sus raíces principalmente en el modelo lineal de consumo en el que se basa nuestro modelo económico. Los bienes son producidos, consumidos y desechados, generando grandes cantidades de residuos durante el proceso [3].

BIN-CT es una aplicación que encapsula un sistema de Inteligencia Artificial (IA) predictiva para la gestión eficiente de la recogida de residuos plásticos, desarrollada dentro del marco del proyecto nacional ECO-IoT (RTC-2017-6714-5). El principal objetivo de la aplicación es responder a las dos grandes preguntas que se plantean a diario los encargados de la recogida selectiva de las ciudades: 1) ¿Qué contenedores debo recoger? 2) ¿En qué orden debo recogerlos?

Al abordar problemas tan complejos como la gestión de los residuos, la eficiencia es uno de los factores clave. No obstante, ni autoridades locales ni entidades privadas suelen invertir suficientes recursos para mejorar la recogida de residuos, donde se suele aplicar siempre el mismo plan estático en la mayoría de casos [4]. Para poder ayudar con esta tarea de una manera eficiente y sostenible, el sistema BIN-CT cuenta con componentes hardware y software. A nivel hardware, disponemos de unos novedosos contenedores sensorizados que emiten datos sobre su localización, temperatura y nivel de llenado. Además, estos contenedores están equipados con una prensa vertical para compactar los residuos plásticos, lo que supone poder extender el tiempo útil entre vaciados.

A nivel software, el sistema dispone de dos módulos principales: 1) Un *Core* interno compuesto principalmente de un Algoritmo de *Deep Learning* para obtener predicciones de llenado diarias para todos los contenedores a partir de los datos de los sensores y un Algoritmo Evolutivo que se encarga de generar rutas eficientes para el vaciado de los contenedores y 2) la aplicación propiamente dicha, que hace de puente entre la información procesada por el *Core* y el usuario. Gracias a las predicciones, el sistema es capaz de conocer por adelantado una estimación del llenado de los contenedores para así decidir cuáles deben recogerse. Esto se realiza mediante un algoritmo de enrutado que determina el orden en el que los contenedores deben ser recogidos.

II. Software Core

Este módulo es el corazón de la aplicación que se ha desarrollado, y la base sobre la que se apoya el sistema completo. Está desarrollado en Maven, que es un *framework* para la realización de proyectos con base en Java.

La primera parte de este *Core* es el módulo de gestión de datos recibidos de los sensores instalados en los contenedores, que se encarga de cargar los datos necesarios al programa desde las bases de datos del sistema. Asimismo, este módulo es responsable también de almacenar los datos procesados por el Core en las bases de datos, según sea necesario. Desde aquí se gestiona una gran variedad de datos de relevancia para la obtención de matrices de distancias y tiempos entre los distintos puntos (nodos) donde están los contenedores desplegados. Dicha información es calculada con la ayuda de la API de Leaflet.

También se procesan datos históricos cada día para su almacenamiento. Esta tarea no es trivial, ya que el dato recibido del sensor no refleja completamente el estado del contenedor asociado. El dato histórico debe reflejar la cantidad aportada de residuo en ese día, lo cuál no es equivalente a tomar la diferencia entre 2 medidas de los sensores volumétricos cada 24 horas. Este dato de cantidad aportada está sujeto a posibles vaciados y compactaciones del contenedor ocurridas en el día, por lo que su cálculo no es tan directo como podría llegar a pensarse. Por tanto, estos eventos son también registrados en las bases de datos para poder obtener información acerca del estado real de los contenedores. De esta manera, nos aseguramos de que el *Core* posee la información más precisa posible para realizar el cálculo de predicciones sobre el futuro nivel de llenado de los contenedores.

Haciendo uso de toda la información recibida y procesada, el *Core* se encarga también de la generación de rutas óptimas para la recogida de residuos. Dichas rutas tienen en cuenta tanto los históricos ya mencionados como las predicciones a la hora de determinar si un contenedor debe ser recogido un día concreto o puede esperar más. Para el trazado de la ruta, hace uso de las matrices de distancia y tiempos. Esta parte del *Core* se ha desarrollado teniendo en cuenta la variabilidad que podrá sufrir la logística en la recogida de residuos a través del tiempo. Por ejemplo, el sistema tiene en cuenta si se dispone de uno o varios vehículos para realizar la recogida, la capacidad de estos y las diferencias de capacidad entre los

contenedores inteligentes, por ejemplo. El módulo también es capaz de adaptar las rutas generadas a diferentes criterios de inclusión/exclusión que le sean indicados.

III. BIN-CT

La aplicación BIN-CT ha sido desarrollada haciendo uso de Angular y NativeScript, utilizando TypeScript como principal lenguaje de programación. Angular es un *framework* de código abierto, desarrollado y mantenido por Google, que nos permite crear una aplicación web adaptable a cualquier tamaño de pantalla de cualquier dispositivo. NativeScript es también un *framework* de código abierto para desarrollar aplicaciones en iOS y Android con Angular. Esta combinación de tecnologías permite realizar un desarrollo común de la lógica de la aplicación, pero adaptando ciertas funcionalidades específicas para distintas plataformas. BIN-CT tiene dos partes diferenciadas: *front-end* y *back-end*.

El *front-end* es la capa de presentación de la aplicación y, por tanto, la parte de esta con la que interactúa el usuario directamente. Los diferentes archivos utilizados para dar forma a la aplicación se encapsulan en componentes, que son elementos que engloban unas características propias e independientes del resto, tanto en apariencia como en comportamiento. Se puede navegar de una interfaz a otra a través de un menú en la parte superior de la misma. En general, cada apartado de nuestra aplicación viene definida en un componente Angular propio. En esta capa se utilizan HTML y CSS para configurar la estructura y apariencia de cada página. La Figura 1 muestra la lista de visualización de clientes, con el botón para desplegar el menú con los diferentes módulos de la aplicación arriba a la derecha.

El *back-end* es un servicio en la nube, desarrollado también en Angular, ayudándonos del módulo TypeORM para facilitar la implementación de la conversión de datos entre el sistema de tipos de nuestro lenguaje de programación y el de la base de datos. Constituye la capa de acceso a datos y funcionalidades de la aplicación. Interpreta las órdenes provenientes del usuario (proporcionadas a través del *front-end*), realiza el procesamiento de datos requerido y devuelve la respuesta correspondiente al usuario a través del *front-end*. Este diseño añade una capa extra de seguridad a la aplicación, al no ser el usuario el que actúa directamente sobre los datos ni sobre la capa que interactúa con la misma. Todas las operaciones del *back-end* están bloqueadas tras una comprobación del token de inicio de sesión excepto el propio intento de inicio de sesión. Si el usuario no está registrado y con una sesión activa, no puede acceder a ninguna de las funcionalidades.

Para hacer funcionar todo el sistema, la aplicación se apoya en diferentes librerías Java: Hibernate para el mapeado de modelos orientados a objetos a bases de datos, Jsprit para la resolución de problemas de enrutado, OkHttp para el manejo de peticiones HTTP y Jackson para convertir texto a JSON y viceversa.

La aplicación permite a nivel usuario la consulta de localización y niveles de llenado de los contenedores en tiempo real, un resumen de niveles de llenado históricos y predicciones



Figura 1: Apartado de listado de clientes.

y rutas de recogidas planificadas para los próximos días. A nivel administrador, la aplicación permite la gestión y consulta de contenedores, sensores y rutas de los vehículos, pudiendo dar de alta a nuevos usuarios y contenedores en el sistema o modificar información referente a ellos. La Figura 2 muestra el apartado de resumen de datos históricos para un contenedor concreto.

En este apartado podemos observar tres gráficas diferentes. La primera de ellas “Evolución de llenado por días”, expone en un diagrama de barras el nivel de llenado del contenedor a las 00:00 de cada día en el rango seleccionado. Además, una línea sombreada refleja la cantidad de residuos (en % sobre la capacidad del contenedor) que fueron introducidos cada día en el mismo. Si se pulsa sobre las diferentes barras, se puede visualizar el número de compactaciones y vaciados realizados en el mismo día.

La segunda gráfica, “Generación de residuos por hora y día”, ofrece más detalladamente el incremento de llenado del contenedor (en %) en intervalos de pocos minutos. El dato representado es el incremento en el intervalo y no el total de llenado del contenedor en el intervalo. La gráfica dispone además de un desplegable y de una leyenda interactiva para mostrar, ocultar y hacer zoom a intervalos concretos a fin de lograr una representación más clara de la información.

La última gráfica, “Evolución de llenado por hora y día”, muestra la evolución del llenado del contenedor (en % sobre la capacidad total) a lo largo del periodo predeterminado, en intervalos de pocos minutos. En esta gráfica son claramente apreciables tanto la velocidad de llenado como las compactaciones y vaciados del mismo.



Figura 2: Menú de visualización de gráficas de datos históricos en tablet apaisada.

IV. Técnicas de IA utilizadas

BIN-CT hace uso de diferentes técnicas de IA. Para las predicciones hace uso de una Red Neuronal Recurrente (RNN), un tipo específico de Red Neuronal Artificial (ANN) basada en *Deep Learning*. Al igual que otras ANNs, este tipo de red está compuesta de múltiples capas ocultas entre las capas de entrada y salida. Las RNNs incorporan conexiones de alimentación y retroalimentación entre capas para identificar dependencias a largo plazo en los datos de entrada. El uso de RNNs nos permite generar predicciones para un set de contenedores utilizando una única RNN en vez de crear y entrenar una para cada contenedor. Esto supone una ventaja puesto que se necesita menos cantidad de datos gracias a la explotación de las similitudes entre contenedores.

Aplicamos también aprendizaje supervisado, que consiste en un proceso iterativo que requiere de un conjunto de datos de entrenamiento (N pares de entrada-salida). Como estamos tratando con predicciones de nivel de llenado, los datos de entrada son los niveles actuales de llenado de los contenedores, y los de salida los niveles para el siguiente día. Es decir, por cada dato de entrada, la ANN produce una salida (posible subida en el llenado) que es comparada con el dato de salida esperado utilizando una función de error. Tras esto, se aplica un proceso para reducir este error actualizando la red hasta que se alcanza cierto criterio de parada [5].

Puesto que entrenar una RNN es costoso (en términos de capacidad computacional) y el número de arquitecturas de RNN es infinita (o extremadamente grande si imponemos restricciones en el número de capas ocultas), estamos forzados a definir una estrategia de búsqueda inteligente para encontrar una RNN óptima. En esta aplicación utilizamos un enfoque de neuroevolución profunda para la generación de predicciones. Nuestro método de optimización se encarga de los siguientes parámetros: el *lookback*, el número de capas ocultas y el número de neuronas por capa oculta. Decidimos entrenar la red utilizando propagación hacia atrás y midiendo la *media del error absoluto* (MAE) de los valores predichos respecto a los observados.

La ecuación 1 representa el problema de encontrar una arquitectura óptima como un problema de minimización, donde N es el número de pares en el conjunto de datos de prueba (X, Y), z_i es el valor predicho de la$i$-ésima muestra, y y_i corresponde con el valor verdadero de la$i$-ésima muestra. Nótese que la RNN es alimentada con datos de predicción $\hat{x}$, y que la arquitectura está restringida por B, H, and L. Para resolver el problema planteado en la ecuación 1 utilizamos un algoritmo neuroevolutivo profundo basado en la $(1+1)$ *Evolutionary Strategy* (ES) [6] y en el *Adam weights optimizer* [7].

$$\begin{aligned} \text{minimizar} \quad & \mathit{fitness} = \frac{1}{N}\sum_{i}^{N} MAE(z_i, y_i) \\ \text{sujeto a} \quad & B \leq \mathit{max\_look\_back} \\ & H \leq \mathit{max\_hidden\_layers} \\ & L \leq \mathit{max\_neurons\_per\_layer} \\ & \hat{x}_i = \begin{cases} x_0 & \text{if } i = 0 \\ z_{i-1} & \text{if } i > 0 \end{cases} \end{aligned} \tag{1}$$

Para poder generar rutas de recogida eficientes utilizamos un Algoritmo Evolutivo (AE) (1+1). El AE utilizado está basado en el principio de ruina y recreación. Es una búsqueda de vecindad amplia que combina elementos de *simulated annealing* y algoritmos *threshold-accepting*. Este enfoque es apropiado para problemas complejos con muchas restricciones [8], [9]. Es una metaheurística multipropósito que puede ser utilizada para resolver diferentes problemas clásicos de enrutado de

vehículos, pudiendo variar las estrategias de búsqueda en función del problema. El Algoritmo 1 muestra el pseudocódigo de este.

Algorithm 1 Pseudocódigo de la generación de rutas.

Entradas: Conjunto de contenedores: (localización, nivel de llenado, predicciones), conjunto de vehículos, matriz de distancias y tiempos, ubicación de la planta de reciclaje
Salidas: Conjunto de listas de contenedores ordenados

```
solucionActual ← solucionAleatoria()
mejorSolucion ← solucionActual
evals ← 0
while evals < totalEvals do
    tareasQuitadas← ruina(solucionActual)
    solucionActual← recreacion(tareaQuitada)
    evaluar(solucionActual)
    if (solucionActual < mejorSolucion) then
        mejorSolucion←solucionActual
    end if
    evals ← evals + 1
end while
return mejorSolucion
```

El algoritmo empieza con una solución inicial aleatoria. Extrae partes de la solución que llevan a un conjunto de tareas que no serán asignadas a un vehículo, obteniendo así la solución parcial. A este proceso se le llama arruinar la solución porque extraemos una parte de una solución. Basándonos en la solución parcial, el algoritmo reintroduce las tareas extraídas llevando a una nueva solución. Este es el proceso de recreación. Si la nueva solución es igual o de mejor calidad que la anterior, se acepta como nueva mejor solución. Estos pasos se repiten hasta alcanzar cierto número de iteraciones.

V. Conclusiones

BIN-CT es una herramienta pensada para su uso en el ámbito profesional en el campo de recogida y reciclaje de residuos. Proporciona a sus usuarios una interfaz simple pero completa, con la que poder monitorizar de manera continua la red de contenedores activa y su contenido, automatizando tareas complejas y costosas como la planificación de rutas eficientes para su vaciado.

La aplicación ha sido desarrollada utilizando Angular y NativeScript, con TypeScript como el principal lenguaje de programación. La aplicación ha sido desarrollada siguiendo un modelo *front-end/back-end*, siendo el *front-end* la parte del sistema con la que interactúa directamente el usuario, es decir, la interfaz, y el *back-end* la parte que engloba tanto la autenticación de usuario como el acceso a las bases de datos y los algoritmos de cálculo de predicciones y rutas.

Con el desarrollo de esta aplicación se pretende dar un paso en pos de la integración de modelos inteligentes sostenibles en nuestras ciudades. Gracias a la ayuda de las técnicas de IA aquí explicadas, se fomenta un cambio en este campo hacia la sostenibilidad económica y ecológica. Por un lado, la utilización de rutas eficientes se traduce en un menor gasto tanto en combustible como en tiempo y personal, lo que supone un ahorro directo en costes de funcionamiento del sistema respecto a uno tradicional. Por otra parte, el ahorro de combustible se traduce en una menor emisión de gases contaminantes por parte de los camiones de recogida, contribuyendo a un ambiente más limpio en nuestras ciudades.

VI. Enlaces

En este enlace estará disponible la descarga tanto de la aplicación como del vídeo demostrativo antes del 7 de Septiembre: https://neo.lcc.uma.es/new/bin-ct-manager/

Referencias

[1] W. Leal Filho, U. Saari, M. Fedoruk, A. Iital, H. Moora, M. Klöga, and V. Voronova, "An overview of the problems posed by plastic products and the role of extended producer responsibility in europe," *Journal of cleaner production*, vol. 214, pp. 550–558, 2019.

[2] I. Manisalidis, E. Stavropoulou, A. Stavropoulos, and E. Bezirtzoglou, "Environmental and health impacts of air pollution: a review," *Frontiers in public health*, vol. 8, 2020.

[3] J. Korhonen, A. Honkasalo, and J. Seppälä, "Circular economy: the concept and its limitations," *Ecological economics*, vol. 143, pp. 37–46, 2018.

[4] S. Oduro-Kwarteng, *Private Sector Involvement in Urban Solid Waste Collection: UNESCO-IHE PhD Thesis*. CRC Press, 2011.

[5] S. Haykin, *Neural networks and learning machines*. Pearson, 2009, vol. 3.

[6] T. Back, *Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms*. Oxford university press, 1995.

[7] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[8] A. Misevicius, "Genetic algorithm hybridized with ruin and recreate procedure: Application to the quadratic assignment problem," in *Knowledge-Based Systems*, vol. 16, no. 5-6 SPEC., 2003, pp. 261–268.

[9] G. Schrimpf, J. Schneider, H. Stamm-Wilbrandt, and G. Dueck, "Record Breaking Optimization Results Using the Ruin and Recreate Principle," *Journal of Computational Physics*, vol. 159, no. 2, pp. 139–171, 2000.

CAEPIA-APP Competition: Saru closet

Sara García-Rodríguez*, Noelia Rico*, Susana Montes†, Irene Díaz*
*Department of Computer Science, University of Oviedo
Emails: [uo258060,noeliarico,sirene]@uniovi.es
†Department of Statistics and O.R. and Mathematics Didactics, University of Oviedo
Email: montes@uniovi.es

Saru closet mobile app is an outfit and closet organizer that also provides the user with personalized recommendations of outfits based on their habits and preferences. The users can introduce all their clothes and accessories using photos, descriptions and information such as style or season. Then, these can be associated with the closet where they are stored, providing useful insight about the location of the items and whether they are available in a concrete moment. The users can also manage outfits, which are made of clothes and accessories, with options for creating them and also to associate them with different tags defined by the own user. Furthermore, these outfits can be connected not only with dates but also with events. Recommendations are given to the users for future events based on prior information.

I. RECOMMENDER SYSTEM

Among all the types of recommender systems [1], *utility based* recommender systems are those that make suggestions to the user based on the information given by the user only. Internally, this information allows the recommender to learn an utility function that rates the possible items to recommend according to the data previously introduced. This type of recommender systems have had a great impact in many different fields in the last decade [2][3][4]. The aim of this application is to learn the patterns that the user follows to wear each outfit in order to give recommendations for concrete events or dates.

One of the main problems of using only the information given by the user is how to gather data enough to create an utility function for each individual user. In this case, the application tries to ensure the engagement of the user for data gathering by its user-friendly interface, which offers an easy environment to register and manage their clothes, closets and outfits. When enough information has been gathered, the users receive recommendations when they assign events to future dates in the calendar.

These recommendations are done by training a model based on different criteria [5]. The output of this model is then combined with a filtering process, just to ensure that all the outfits recommended are suitable for a concrete moment of time. For example, maybe some of the outfits recommended contain clothes that are not available and, depending how close the date of the event is, these can be discarded or at least the user can receive a warning about the clothes being unavailable (for example, because they are in the washing machine).

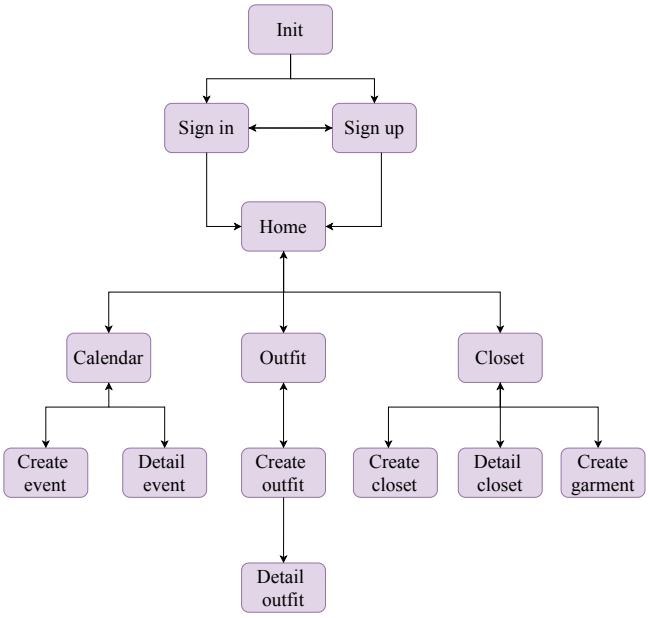


Fig. 1: Navigation map between the different of the app.

Once the user receives a recommendation, this can be accepted or discarded. These actions are also registered to evaluate the performance of the model. In order to evaluate the satisfaction of the user, when the recommendations are discarded a concrete amount of times, the current models are removed. Then, new models are trained and therefore the recommendations are updated, seeking to improve the patterns learnt in order to provide better recommendations to the user.

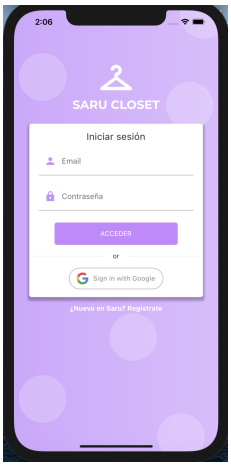
II. USABILITY

The functionality of the app can be mainly divided into management of clothes, closets, and outfits. The app provides CRUD (Create, Read, Update and Delete) operations for all these different objects. It can be noticed that, intuitively, these objects are clearly associated among them, as clothes are stored in closets and outfits are just combinations of clothes. The app also can be used as a tool for knowing the state of the clothes, for example if they have been used and therefore they are in the washing machine, in which case they would not be available to wear so the user could not select outfits containing them.

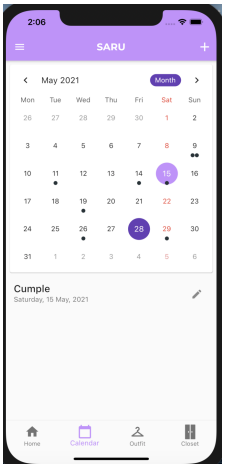
Different CRUD operations are presented in different screens. A diagram indicating how the screens interact and how they can be accessed guiding the workflow of the user interaction with the app is shown in Fig. 1. The user opens the app in an initial screen (shown in Fig. 2a), which gives access to additional screens to sign in or sign up by means of a traditional log in introducing data or using a google account. Once the user has log in, a menu bar is presented, which allows to change from one screen to another in a straight-forward way in order to perform different operations. These operations can be divided into the ones related with the calendar, the ones related to outfits, and the ones related to closets, as shown in the fourth row of the diagram. Different closets can be created and inside them the user can introduce new clothes. From the closets' screens the user can also check the information about the clothes and manage where these are stored and their state. The screens focused on the outfits allow to use the list of clothes in the closets to create new outfits. These outfits can also be associated with events in the calendar. On the one hand, the events related to past dates show the historic data archived for the user. On the other hand, new events for future dates allow to obtain a recommendation made by the app.

Some screenshots of different parts of the application are shown in Fig. 2. The idea is to organize and present the functionalities of the application to the user in a way that is similar to the one currently provided by many popular clothing stores. By doing this, the time required for first-time users in order to get familiar with the app decreases and the chances of engaging the user with the use of the app increase. This fact is important, as ensuring this engagement from the beginning is a key part of the process in order to collect data until enough information is gathered so recommendations can be made. Also, the interface has been designed following the convention of usability recommended for mobile apps [6][7][8].

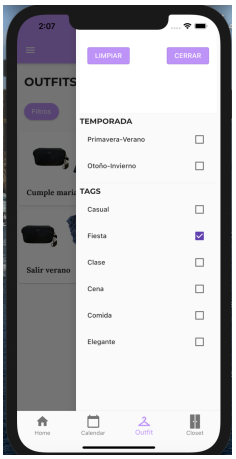
These screenshots shown in Fig. 2 summarize some parts of the interface of the app. Fig. 2a shows the initial screen that the user is presented when opening the app. From the calendar option shown in Fig. 2b, the user is able to explore the history by checking past dates and also to plan an event or explore past ones. The recommendations are associated with future events. In this screenshot can be also observed the menubar, that allows to access the outfits and closets tabs quickly. Screenshots showing how filters for outfits and clothes inside the closets can be used are shown in Fig. 2c and Fig. 2d. The creation of an outfit allows to pick up the list of clothes and a selection of tags associated and also other information. These tags are later used by the model trained to provide the user with custom recommendations, and also may change at any moment of the lifetime of the outfit. An outfit and all its associated information can be edited from a screen as the one shown in Fig. 2e. Also, for a concrete garment, all the outfits registered are shown in a form of a list of carousel panels that allows to compactly visualize the possible ways in which it can be combined. This allows the user to quickly explore suitable combinations in case of express interest in wearing a concrete garment, as shown in Fig. 2f.



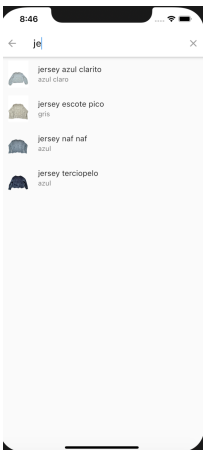
(a) Sing in and sign up with email or Google account.



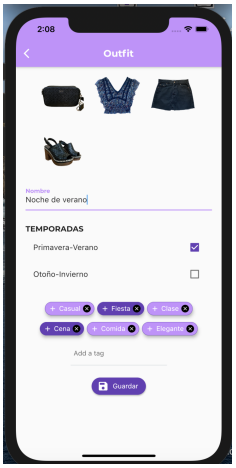
(b) Calendar for registering outfits and events and get recommendations.



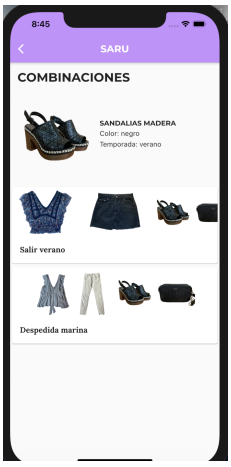
(c) Sidebar for filtering the outfits by tags of season.



(d) Filter to find the clothes in the closet faster.



(e) Details of an outfit and edition panel.



(f) Carousels of outfits associated with a concrete garment.

Fig. 2: Screenshots of *saru closet* app.

III. IMPLEMENTATION

The app has been developed in *flutter*,[1] a Google's UI toolkit for building mobile applications (and also web, desktop, and embedded devices) that are natively compiled to any device from a single codebase [9]. This allows to create the app from one single code, compiling it natively into Android and iOS, the most common popular mobile operating systems. The election of using flutter has been motivated precisely by the fact that the app is natively compiled, which makes it being optimized for a specific platform in terms of speed and responsivity.

All the data provided by the user is stored using the *Firebase Realtime Database*[2]. This is a NoSQL cloud-hosted database provided by Google that stores and sync data in realtime [10]. Data is stored as `JSON` collections and synchronized with the app clients, that automatically receive updated data when the app is accessed.

The app needs to know the identity of the users in order to provide them with recommendations. Another important aspect is to ensure that the data is safely accessed by the user that introduces it. Therefore, knowing the user's identity is a key point to allow the app to securely save user data in the cloud. These two aspects are covered by *Firebase Authentication*[3], that gives backend services as well as SDKs and UI libraries to authenticate the app's users. These services are used in the app, which supports authentication using their or email and password or Google, allowing to sign in with the gmail account without further registration.

The model for the recommendation is done with help of *TensorFlow*[4], which is an end-to-end open source platform for machine learning that allows developers to easily build and deploy machine learning powered applications. This provides a library for compiling models to create recommender systems[5]. Another tool provided by this framework is the specification *TensorFlow Lite*[6] to deploy machine learning models on mobile devices. This is an open source deep learning framework for on-device inference.

The interaction of the user with the app must be registered in order to have feedback of how appropriate the recommendations given to the user are, which allows to keep track the performance of the models in order to maintain them if they have a good performance or modify them in case the performance is poor. To this end, Google Analytics[7] is used, as it allows to track the response given by the user and provides libraries for its integration with all the tools mentioned above.

The code of this app can be in found in the Github repository https://github.com/saragarciarg/saru-closet, as well as the link to a video presentation and detailed guides for its installation.

[1] https://flutter.dev/
[2] https://firebase.google.com/docs/database
[3] https://firebase.google.com/docs/auth
[4] https://www.tensorflow.org
[5] https://www.tensorflow.org/recommenders
[6] https://www.tensorflow.org/lite
[7] https://analytics.google.com/

IV. CONCLUSION

The *Saru closet* app presented in this work provides a tool for clothes, closets and outfits management that has been developed using the latest technologies available. The aim of this app is to simplify the process of a daily task such as outfit decision, engaging the user by means of the app's intuitive screen and flexible use. Furthermore, the app uses machine learning techniques in order to give recommendations to the user based on the information previously registered, which helps the user to select an outfit according to the previous shown preferences.

REFERENCES

[1] Charu C Aggarwal. *Recommender Systems*. Springer International Publishing, Cham, 2016.
[2] Shiu-Li Huang. Designing utility-based recommender systems for e-commerce: Evaluation of preference-elicitation methods. *Electron. Commer. Res. Appl.*, 10(4):398–407, 2011.
[3] Morteza Zihayat, Anteneh Ayanso, Xing Zhao, Heidar Davoudi, and Aijun An. A utility-based news recommendation system. *Decis. Support Syst.*, 117:14–27, 2019.
[4] Alexander Felfernig, Seda Polat-Erdeniz, Christoph Uran, Stefan Reiterer, Muesluem Atas, Thi Ngoc Trang Tran, Paolo Azzoni, Csaba Kiraly, and Koustabh Dolui. An overview of recommender systems in the internet of things. *J. Intell. Inf. Syst.*, 52(2):285–309, 2019.
[5] Yong Zheng. Utility-based multi-criteria recommender systems. In *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing - SAC '19*, New York, New York, USA, 2019. ACM Press.
[6] Teuta Cata and Ben Martz. Comparing mobile apps usability characteristics for designers and users. *Journal of International Technology and Information Management*, 24(4):4, 2015.
[7] Xiaoxiao Ma, Bo Yan, Guanling Chen, Chunhui Zhang, Ke Huang, Jill Drury, and Linzhang Wang. Design and implementation of a toolkit for usability testing of mobile apps. *Mobile Networks and Applications*, 18(1):81–97, 2013.
[8] Christopher Vendome, Diana Solano, Santiago Linán, and Mario Linares-Vásquez. Can everyone use my app? an empirical study on accessibility in android apps. In *2019 IEEE International Conference on Software Maintenance and Evolution (ICSME)*, pages 41–52. IEEE, 2019.
[9] Marco L Napoli. *Beginning flutter: A hands on guide to app development*. Wiley-Blackwell, Hoboken, NJ, 2019.
[10] Houssem Yahiaoui. *Firebase Cookbook*. Packt Publishing, Birmingham, England, 2017.

XIX Conferencia de la Asociación Española para la Inteligencia Artificial

Competición de Vídeos de IA Divulgativos



Competición Vídeo: Circuitos que evolucionan

Sara Bueno Albacete
Resilient Works
Email: sbueno.albacete@gmail.com

Federico Castejón Lapeyra* Enrique J. Carmona Suárez†
Departamento de Inteligencia Artificial
Escuela Técnica Superior de Ingeniería Informática
Universidad Nacional de Educación a Distancia (UNED)
Email: *federico.castejon@correo.gob.es †ecarmona@dia.uned.es

***Abstract*—En este artículo, se presenta un video que explica a niños y jóvenes cómo se pueden utilizar técnicas de computación evolutiva para que un ordenador diseñe circuitos electrónicos. Este vídeo fue creado para la Semana de la Ciencia 2020 y, por lo tanto, se presenta a la competición de videos de AEPIA, dentro de la categoría D, videos ya existentes. El uso de la animación, como lenguaje visual, resulta muy atractivo para el público objetivo, formado por niños y jóvenes, así como facilita mucho la comprensión de conceptos que, de otra manera, pueden resultar complejos. Al tratarse de un algoritmo inspirado en la evolución biológica, se planteó utilizar la teoría de Darwin y la evolución como hilo conductor. Además, se presentó el concepto de circuitos como seres vivos, para buscar que el público joven empatizara con los mismos, así como hacerlos, no sólo simples circuitos, sino personajes de una historia. En visualizaciones informales por niños y adolescentes, se comprobó que el video fue muy bien recibido.**

I. Introducción

Este video explica a niños y jóvenes cómo se pueden utilizar técnicas de computación evolutiva para que un ordenador diseñe circuitos electrónicos. El video se presenta a la competición de videos de AEPIA, dentro de la categoría D, videos ya existentes.

El artículo está estructurado tal y como se indica a continuación. En primer lugar, se proporciona el enlace al video. En segundo lugar, se aborda la descripción de las herramientas y materiales empleados. En tercer lugar, se indica la justificación del contenido. En cuarto lugar, se describe someramente la técnica de computación evolutiva descrita en el video. En quinto lugar, se describe la metodología utilizada para el desarrollo del video. En sexto lugar, se indican las reacciones al video en visionados informales. Finalmente, se recopilan las conclusiones.

II. Enlace al video

El video desarrollado se puede encontrar en el siguiente enlace de la plataforma Youtube:
https://youtu.be/_QIkV7vAahA

III. Descripción de las herramientas y materiales empleados

En este apartado se indican las herramientas utilizadas para el desarrollo del video. La animación se desarrolló mediante el programa de animación 2D Anime (Moho) v10. El audio se capturó y editó con el programa de software libre Audacity v2.4.2. Finalmente, el video se montó con el programa de edición de video Cyberlink PowerDirector 16. Adicionalmente, se utilizaron herramientas de Microsoft Office 365 tales como PowerPoint o Microsoft Word.

Todos los materiales utilizados en el video, salvo los indicados posteriormente, se desarrollaron a medida para el mismo, incluyendo el *concept art* de los personajes y su animación, así como el guión y la voz en *off*. Por otro lado, se han utilizado efectos de sonido y música, libres de derechos, obtenidos de Zapsplat [1] y una imagen de Charles Darwin, libre de derechos, obtenida de Pixabay [2].

IV. Justificación del contenido

Este vídeo fue creado para la Semana de la Ciencia 2020, dentro de la primera jornada científica de la Escuela Internacional de Doctorado de la UNED. Este evento se celebró con el propósito de acercar a la comunidad científica y a la sociedad los trabajos en investigación, innovación y transferencia que se realizan en la Escuela. El video está inspirado en una serie de trabajos [3], [4], [5], [6] llevados a cabo por miembros del grupo SIMDA[1] de la UNED, en los que se propone el uso de algoritmos evolutivos para el diseño automatizado de circuitos electrónicos analógicos, como asistencia ante la falta de herramientas completamente automatizadas en dicho campo.

V. Técnica de Inteligencia Artificial descrita en el video

Desde los años 70, los circuitos digitales han venido reemplazando a los analógicos de forma general, si bien, hay áreas en las que los circuitos analógicos siguen siendo irreemplazables. En el diseño analógico no existe una metodología general ni existen herramientas automatizadas tan avanzadas como en el ámbito digital. Esto se debe a la mayor complejidad del diseño analógico, que requiere, todavía, una gran implicación por parte del experto. Por esta razón, hay un gran interés en disponer de herramientas automáticas para facilitar este tipo de diseños.

En el ámbito de la electrónica evolutiva existen trabajos previos basados en diferentes algoritmos. Básicamente, el problema del diseño analógico consiste en sintetizar un circuito electrónico analógico que cumpla un conjunto de requisitos. Esto se consigue abordando dos tareas: selección de la topología del circuito (determinación de los componentes y sus interconexiones) y dimensionamiento (determinación de los valores de dichos componentes).

[1] http://simda.uned.es

En los trabajos que son fuente de inspiración de este video [3], [4], [5], [6], se ha buscado obtener resultados competitivos en el problema del diseño automático de circuitos electrónicos analógicos, en relación con resultados previos obtenidos en la literatura y, para ello, se han propuesto dos algoritmos. El primero está basado en evolución gramatical (EG) y, el segundo, en una nueva variante de EG, denominada evolución multigramatical (EMG). EG es una nueva variante de programación genética basada en el uso de cromosomas formados por cadenas binarias de longitud variable que se decodifican mediante una gramática libre de contexto. EMG es una variante de EG en la que se utiliza un enfoque modular, dividiendo el problema en subproblemas, cada uno de los cuales, se aborda definiendo una gramática particular. En concreto, para resolver el problema planteado, se utiliza una gramática para cada tarea del diseño de circuitos: una para la selección de la topología y otra para el dimensionamiento.

Las dos metodologías descritas en los trabajos de referencia se han aplicado a la síntesis de siete circuitos de *benchmarking* utilizados en artículos relevantes de la bibliografía, obteniendo resultados competitivos con EG y una importante mejora de estos resultados con EMG. Adicionalmente, se consiguió una reducción notable del número de componentes de los circuitos obtenidos.

VI. Metodología

El video desarrollado se ha basado en animación para ilustrar los conceptos que se iban a explicar mediante una voz en *off*. El uso de la animación, como lenguaje visual, resulta muy atractivo para el público objetivo, formado por niños y jóvenes, así como facilita mucho la comprensión de conceptos que, de otra manera, pueden resultar complejos. Por ejemplo, los conceptos como la recombinación del ADN, la transmisión de caracteres de padres a hijos y la selección natural, pueden usarse computacionalmente para producir soluciones a un problema, como es el caso del problema del diseño de circuitos.

Se decidió partir del objetivo que se quiere conseguir (diseño de un circuito) y explicar cómo se había conseguido, pues pensamos que esta forma resultaba más clara para el público. Al tratarse de un algoritmo inspirado en la evolución biológica, se planteó utilizar la teoría de Darwin y la evolución como hilo conductor. Además, se presentó el concepto de circuitos como seres vivos, para buscar que el público joven empatizara con los mismos. Por ello, los circuitos se presentaron de una forma humanizada, incorporando ojos, patas y brazos, como elementos que los humanizan y que, así, pueden mostrar emociones a través de los mismos. Por ejemplo, los circuitos pueden mostrar ansiedad antes de ser seleccionados, sufrir alegría o frustración, etc. Con todo ello, se trata de hacerlos, no sólo simples circuitos, sino personajes de una historia. De esta manera, se buscaba una mayor comprensión del mecanismo de algoritmo evolutivo, basado en la vida en sí. Así, aparecen conceptos como el de un circuito y una «circuita», así como el mecanismo de selección natural utilizando un circuito «contrahecho», para buscar que la audiencia empatice con el mismo y que, finalmente, resulta ser el que mejor rendimiento proporciona.

Dado que uno de los circuitos a diseñar es un sensor de temperatura, el concepto de adaptación al medio se plantea como un desierto donde aquellos que no están bien adaptados mueren. El concepto de selección, que se veía complejo de explicar conceptualmente, se describió como un concurso con puntuaciones, que serían equivalente a los valores de adaptación usados por el algoritmo. De esta manera, se consigue mostrar de una forma muy visual, la adecuación de un circuito candidato.

Por último, se vuelve al plano real, para explicar las capacidades de este tipo de algoritmos, lo que se ha logrado hacer con ellos y, así, motivar a la audiencia a participar en el desarrollo o evolución futura de este campo.

El proceso de desarrollo del video contempló los pasos que se indican a continuación:

1) Se seleccionaron los conceptos a tratar en el video y se redactó una explicación lo más clara y sencilla posible de los mismos. Con esta información se preparó una sinopsis del video.
2) Se preparó un *storyboard* con la secuencia de la narrativa del video, ordenada de menor a mayor dificultad de conceptos básicos y, posteriormente, los conceptos relacionados. Este *storyboard* se preparó como una presentación en PowerPoint. En cada diapositiva del mismo se utilizó una imagen ilustrativa del concepto a explicar y el texto para la voz en *off*. La imagen ilustrativa tenía carácter temporal y se sustituiría, posteriormente, por las animaciones. La numeración de cada diapositiva se utilizó de forma exhaustiva en el desarrollo del video para facilitar el montaje del mismo.
3) Se procedió a grabar las locuciones de la voz en *off*, utilizando el programa Audacity. En este programa se realizó un procesado del audio para la limpieza del mismo, así como una compresión y normalizado del mismo.
4) Se montó un esqueleto del video con la herramienta PowerDirector, marcando la duración mediante el audio grabado e incorporando las imágenes ilustrativas utilizadas en el PowerPoint.
5) Se desarrollaron los dibujos básicos de los personajes, donde se decidió hacer una asociación entre los seres vivos y los circuitos. El concepto de los circuitos en sí y las moléculas de ADN presentaron la mayor dificultad para su dibujo.
6) Se desarrollaron las animaciones para cada segmento del video con el programa Anime, que se incorporaron a la herramienta PowerDirector. La animación de los personajes del video tales como, las moléculas de ADN, los esqueletos y, sobre todo, los circuitos requirió un trabajo especialmente laborioso, utilizando el concepto de huesos que, una vez configurados, permitió mover los personajes y programar sus movimientos en una secuencia.

7) Se añadieron efectos de sonido para complementar aquellos puntos de la animación que lo requirieran, así como una banda sonora.
8) Finalmente, se generó el video completo con todos los componentes.

VII. Discusión

Una vez desarrollado el video, se mostró, de forma informal, a niños y adolescentes, a los que se preguntó por su opinión. En general, el video fue muy bien recibido, destacando que lo que más les había gustado eran los esqueletos de la secuencia de la adaptación al medio en el desierto, así como, su identificación o empatía con el circuito «contrahecho», que es algo buscado en el guión del video. Por otro lado, hasta los niños más pequeños identificaban elementos tales como darse la mano y los corazones, como formar una familia y tener niños, lo cual nos alegraba, pues nos preocupaba que no se entendiera.

Por otro lado, también se ha preguntado, informalmente, a adultos, destacando que les parecía muy sorprendente que se pudieran diseñar circuitos complejos a partir de una metodología aparentemente sencilla. También, mostraron una gran curiosidad por el tema, preguntando si el algoritmo se podía aplicar para otros tipos de problemas e, incluso, proponiendo nuevos problemas.

VIII. Conclusiones

La competición de videos en el ámbito CAEPIA-2021 muestra la importancia que está adquiriendo el uso del video como recurso para la divulgación de las técnicas en Inteligencia Artificial, dentro de la divulgación de la ciencia y la investigación científica con carácter más general.

En este artículo, se ha presentado un video que explica a niños y jóvenes cómo se pueden utilizar técnicas de computación evolutiva para que un ordenador diseñe circuitos electrónicos.

El uso de la animación, como lenguaje visual, resulta muy atractivo para el público objetivo, formado por niños y jóvenes, así como facilita mucho la comprensión de conceptos que, de otra manera, pueden resultar complejos.

En visualizaciones informales por niños y adolescentes, se comprobó que el video fue muy bien recibido.

References

[1] Zapsplat, "Free sound effects & royalty free music," Recuperado de https://www.zapsplat.com, 2020.
[2] Pixabay, "Stunning free images & royalty free stock," Recuperado de https://pixabay.com, 2020.
[3] F. Castejón, "Diseño automático de circuitos electrónicos analógicos mediante algoritmos evolutivos," Ph.D. dissertation, 2020.
[4] F. Castejón and E. J. Carmona, "Introducing modularity and homology in grammatical evolution to address the analog electronic circuit design problem," *IEEE Access*, vol. 8, pp. 137 275–137 292, 2020.
[5] F. Castejón and E. J. Carmona, "Automatic design of analog electronic circuits using grammatical evolution," *Applied Soft Computing*, vol. 62, pp. 1003 – 1018, 2018.
[6] F. Castejón and E. Carmona, "Automatic design of electronic amplifiers using grammatical evolution," in *Actas de Multiconferencia CAEPIA-13*, A. Alonso-Betanzos *et al.*, Eds., 2013, pp. 703–712.

Competición Vídeo: NAO, el robot cuentacuentos

Mario Álmagro Cádiz
Departamento de Lenguajes y Sistemas.
ETSI Informática.
Universidad Nacional de Educación a Distancia (UNED)
Juan del Rosal 16, 28040-Madrid
Email: malmagro@lsi.uned.es

Víctor Fresno Fernández
Departamento de Lenguajes y Sistemas.
ETSI Informática.
Universidad Nacional de Educación a Distancia (UNED)
Juan del Rosal 16, 28040-Madrid
Email: vfresno@lsi.uned.es

Félix de la Paz López
Departamento deInteligencia Artificial
ETSI Informática
Universidad Nacional de Educación a Distancia (UNED)
Juan del Rosal 16, 28040-Madrid
Email: delapaz@dia.uned.es

***Resumen*—La generación inteligente de gestos en robots humanoides para la interacción con humanos es un problema abierto. Actualmente, las técnicas basadas en la correspondencia simple entre gesto y palabra resultan tediosas pues acotan mucho la reutilización de gestos para palabras semánticamente similares. Ademas, coreografiar nuevos gestos siempre es una tarea pesada para el programador. En este vídeo, vamos a enseñar como utilizar técnicas de procesado del lenguaje natural (PLN) para el disparo inteligente de gestos utilizando la similaridad semántica. En particular, lo vamos a ejemplificar con un robot humanoide (NAO) que cuenta un cuento para niños.**

I. Introducción

La interacción entre humanos y robots (HRI) es un área de interés creciente en la Inteligencia Artificial que pretende hacer más natural la interacción con los robots. En este sentido, han aparecido numerosos estudios de investigación sobre las interacciones verbales y visuales con los robots. El presente trabajo[1] se centrará en la comunicación no verbal y, más concretamente, en los gestos relacionados con el habla, que es una cuestión abierta. Para ello se utiliza un robot humanoide NAO, que está diseñado específicamente para la interacción social con humanos.Visite el vídeo de esta investigación en https://bit.ly/2MPduLv.

II. Herramientas y materiales utilizados

Para desarrollar este trabajo, hemos utilizado un robot humanoide NAO de la empresa SoftBank, ayudados por el software de desarrollo Choregraphe para elaborar los gestos. Los algoritmos de procesamiento del lenguaje natural (PLN) se han implementado en C++ standard.

III. Técnicas de Inteligencia Artificial utilizadas

Con el fin de desarrollar esta parte de la interacción humano-robot o HRI, se propone una nueva arquitectura para la asignación de gestos al habla basada en el análisis de las similitudes semánticas. De este modo, los gestos se seleccionarán de forma inteligente mediante técnicas de PLN. Las

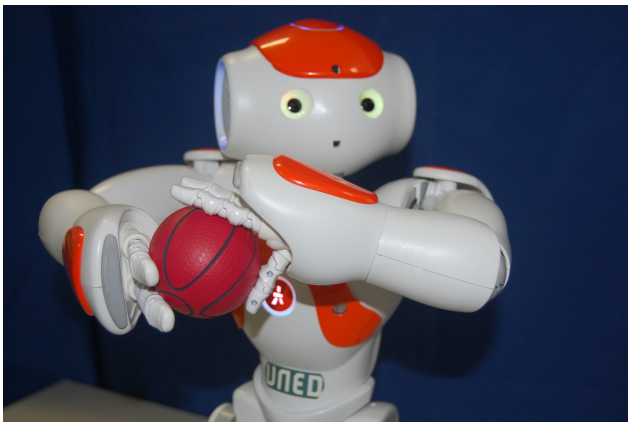


Figure 1. Robot humanoide N AO

condiciones de selección de gestos se determinarán a partir de una evaluación de la eficacia de diferentes modelos lingüísticos en una tarea de sustitución léxica aplicada a la anotación de gestos. A partir de este análisis, se pretende comparar los modelos basados en el conocimiento experto y los modelos estadísticos generados a partir del aprendizaje léxico.

El prototipo descrito tiene como objetivo ofrecer una interacción más fluida e inteligente, acompañando el discurso con los gestos adecuados. En este ámbito, la línea de actuación general ha sido el uso de sistemas basados en reglas, vinculando manualmente palabras claves a gestos predefinidos. Sin embargo, la evolución de las interfaces robóticas requiere sistemas más generales y dinámicos. Nuestra propuesta aborda la asociación de partes del discurso del robot a gestos predefinidos de forma automática. Sin embargo, no se limita a buscar las palabras clave que se hayan establecido como disparadores de los gestos, sino que por medio de técnicas de estimación de similitud semántica es capaz de encontrar en el texto términos relacionados con las palabras establecidas como disparadores de gestos. En particular, hemos identificado cada

gesto con un par de palabras representativas para caracterizar sus significados y, de esta forma, procesar tanto texto como movimientos de forma homogénea. Como resultado, el robot es capaz de lanzar gestos a partir de palabras no vistas anteriormente ni asociadas a ningún gesto. La representación se ha basado en *«word embeddings»*, que son vectores de palabras en espacios vectoriales donde se conserva una isometría entre el espacio de representación vectorial y el espacio semántico, y generados por modelos pre-entrenados, calculando la similitud entre gestos y palabras, e identificando el gesto más apropiado dado un fragmento de texto.

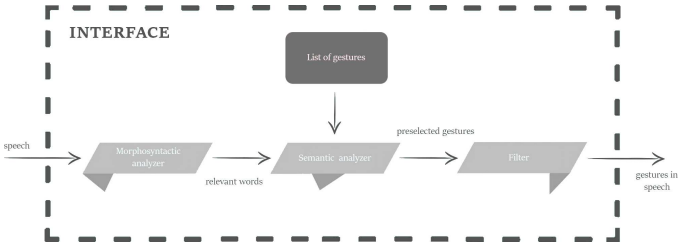


Figure 2. Resumen de la arquitectura

IV. Recursos narrativos y estéticos

En vídeo se desarrolla en tres fases, cada una asignada a uno de los autores, más una conclusión. Aunque se exponen conceptos técnicos, se intenta siempre explicarlos con sentido didáctico para hacerlo entendible por un amplio espectro de público. En la primera fase, el Prof. Félix de la Paz introduce el concepto de interacción humano robot y de sus posibles usos presentes y futuros. En la segunda fase, el Prof. Víctor Fresno explica de manera genérica en qué consisten las técnicas de procesamiento de lenguaje natural y sus posibles usos y aplicaciones.En la tercera fase, el Prof. Mario Almagro explica la aplicación concreta que se ha desarrollado para disparar gestos en el robot NAO basados en las técnicas de PLN explicadas. Por último se hacen algunas reflexiones sobre el alcance del experimento y el trabajo que queda por realizar.

V. Usos didácticos

El vídeo está dirigido a cualquier público con inquietudes científicas. Se intentan explicar todos los conceptos con un lenguaje claro y sencillo, comenzando con los temas de menor a mayor complejidad, siempre manteniendo un objetivo más didáctico que estrictamente científico. Según nuestro criterio, pensamos que sería ideal para captar el interés científico de estudiantes de últimos cursos de Bachillerato. La inteligencia artificial y la robótica son dos áreas muy atractivas y de gran actualidad dentro de la ingeniería que necesitan atraer el talento para formar profesionales en los próximos años para cubrir la demanda que se está produciendo ya.

VI. Results

La generación inteligente de gestos utilizando técnicas de PLN se ha demostrado útil para el rápido prototipado de aplicaciones de HRI, al reducir drásticamente el tiempo de desarrollo que se invierte al coreografiar los gestos. Estos gestos a veces, en este trabajo, pueden resultar poco aproximados al concepto que los dispara, pero esto es simplemente consecuencia de que sólo se ha trabajado con una base de 200 gestos. Aumentar esta base de gerstos redundará, sin duda, en una correspondencia más representativa entre gesto y palabra.

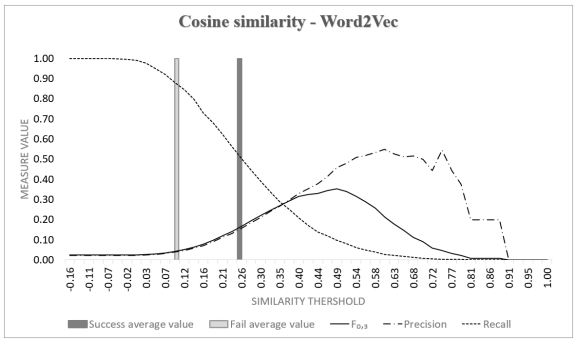


Figure 3. Resultados experimentales

Agradecimientos

Queremos agradecer especialmente al Centro de Medios Audiovisuales (CEMAV) de la UNED por la realización y dirección del vídeo y en especial a Raquel Viejo, Juan Manuel Segura, Antonio Fernández, Pilu Blasco y Yolanda Moreno. Gracias por hacernos tan fácil el trabajo.

Referencias

[1] M. Almagro, V. Fresno, and F. de la Paz, “Speech gestural interpretation by applying word representations in robotics,” *Integrated Computer-Aided Engineering*, vol. 26, no. 1, pp. 97–109, 2019.

XIX Conferencia de la Asociación Española para la Inteligencia Artificial

Doctoral Consortium



Soft Computing and Artificial Intelligence for Interpretable Analysis of Dermoscopic Images*

*Note: This thesis started on July 1, 2018.

Lidia Talavera-Martínez

Advisors: Manuel González-Hidalgo and Pedro Bibiloni
SCOPIA Research Group. University of the Balearic Islands, E-07122 Palma, Spain
Health Research Institute of the Balearic Islands (IdISBa), E-07010, Palma, Spain
l.talavera@uib.es

I. Summary

My research aims to develop an innovative and explainable skin lesion's diagnostic aid tool that can be interpreted by the specialists clinicians, in such a way that it not only determines whether there is a malignant lesion or not but also what such classification is due to, helping thus to a better help diagnosing of dermal lesions and subsequent treatment of the patient.

To this end, we propose a multitasking CAD tool using CNNs that integrates into a complete system the different criteria that make up the diagnosis protocols applied by the physicians, such as the symmetry of the lesion, the presence of dermoscopic structures (presence of streaks, pigment networks, dots/globules...), as well as other well-established tasks as the lesion segmentation and classification, to assess the malignancy of a skin lesion, see Figure 1.

The main objectives in which the research project of the doctoral thesis are broken down below:

1) Relate the fields of computer vision and dermatology. In particular, make a correspondence between the techniques of artificial intelligence and Soft Computing in the analysis of dermoscopic images; and the indicators of interest in clinical practice and the dermatological nomenclature associated with them. The aim is to address the following questions: Are they very divergent fields of knowledge? What information is of interest and can it also be extracted automatically?
2) Design and implement a diagnostic aid system based on dermoscopic images. The applied character is emphasized: to be used as such, it must be easily adopted by the specialist, so it must be simple, reliable, and effective. It is intended to answer the following question: To what extent is it possible to take advantage of the latest developments in computer vision for dermatological image analysis?
3) Validate the diagnostic aid system. In a first approach to its transfer to clinical practice, a quantitative validation of its results is proposed, and a qualitative validation of its ease of use and adoption, addressing the following questions: Is it possible to consider the transfer to clinical practice of this system? To what extent do validation and interpretability contribute to increasing trust in the system?

Ultimately, the project comprises some theoretical aspects and the implementation of the algorithms.

This work was partially supported by the Spanish Grant FEDER/Ministerio de Economía, Industria y Competitividad - AEI/TIN2016-75404-P. Lidia Talavera-Martínez also benefited from the fellowship BES-2017-081264 conceded by the *Ministry of Economy, Industry and Competitiveness* under a program co-financed by the *European Social Fund*.

II. Methodology and Work Plan

The methodology in our case is fundamentally based on the continuous study of the literature, the analysis to determine which problems are technically viable, but have not yet been solved, and the scientific rigor when carrying out the research with which we intend to solve them.

We outline the following series of tasks that I faced during my doctoral thesis and that have helped, until now, to fulfill in the best possible way the three objectives mentioned above.

A. Computational Texture Features of Dermoscopic Images and Their Link to the Descriptive Terminology - A Survey

There are various published reviews addressing diverse aspects of melanoma. To our knowledge, all of these surveys do not focus on the texture analysis in melanomas, but they mention it at a glance. This is the case of Korotkov *et al.* [1] and Pathan *et al.* [2], whose reviews summarize, compare and classify methods and studies with respect to a wide variety of aspects, such as imaging techniques, melanoma diagnosis methods, image pre-processing, lesion border detection, feature extraction and selection, and lesion classification. Also, Oliveira *et al.* [3] focus on unifying the different segmentation methods of melanoma.

Thereby, we proposed and carried out a study focused on establishing the relationship between the characteristics that can be extracted automatically using image processing and analysis, with the descriptive terminology and the rules used by specialists to establish their diagnosis. For this, more than

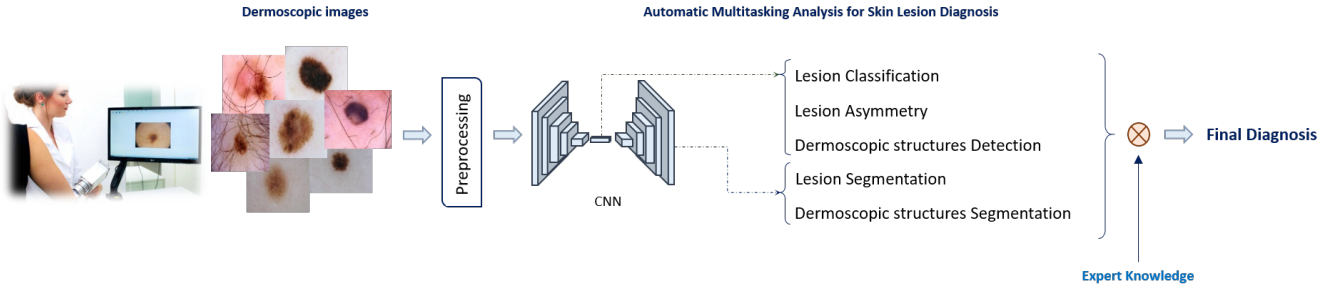


Fig. 1: Visual summary of the main goals of the research project of the doctoral thesis.

100 articles were reviewed, in which we have dealt with the following aspects:

- The particularities and characterization of dermoscopic images. As well as the descriptive terminology used in dermatological clinical practice.
- The state of the art of Artificial Intelligence and Soft Computing techniques for the analysis of dermoscopic images and specific techniques to extract the characteristics of interest.
- The diagnostic rules developed and applied by specialists in clinical practice.

The result of this work was compiled and published in an impact journal within the first quartile [4], thus considering the first objective of our research plan fully accomplished. It is worth noting that the current research must continue to be updated since the subject matter of the doctoral thesis is an area of hot research in constant evolution. The continuous appearance in the literature of new techniques of interest rewrite the state of the art and makes the continuous collection and analysis of the bibliography essential.

B. Comparative Study of Dermoscopic Hair Removal Methods

When analyzing skin cancer in dermoscopic images, the hairs and their shadows on the skin may occlude relevant information about the lesion at the time of diagnosis, reducing the ability of automated classification and diagnosis systems. To tackle this situation, we initially conducted a comparative study of six state-of-the-art hair removal algorithms, each with a different approach to segment and inpaint pixels in hair regions. Currently, there are no datasets that contain the same images with and without hair, which is necessary to quantitatively evaluate the algorithms. Thus, we simulate the presence of hair in hairless images extracted from publicly known datasets, see Figure 2. Then, the results obtained with the hair removal algorithms can be contrasted with the reference hairless images. To quantitatively evaluate its effectiveness, we used a series of performance measures that evaluate the similarity between the original hairless image and that obtained by each of the algorithms. Finally, we use a statistical test to verify the superiority of one method over the others. This work was presented at the VipIMAGE 2019 congress [5].

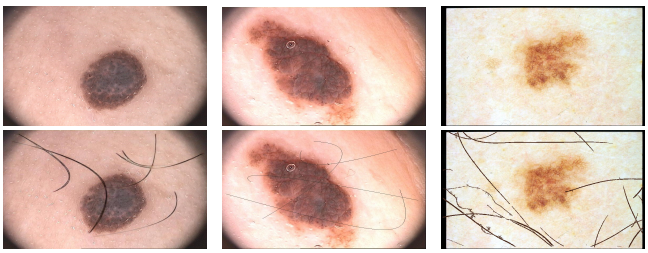


Fig. 2: Original images (top) and simulated hair images (bottom), respectively by a deep neural network [6] (left), “HairSim” [7] (middle), superimposing a hair mask [8] (right).

C. Hair Segmentation and Removal in Dermoscopic Images using Deep Learning

As demonstrated in the prior subsection, extensive research has been done addressing the hair removal process in dermoscopic images. To the best of our knowledge, previous works presented approaches based on traditional computer vision techniques.

In order to address this issue in more depth, we also designed a new approach for this task based on deep learning techniques. As can be seen in Figure 3, our proposed model relies on an encoder-decoder architecture, with convolutional neural networks, for the detection and posterior restoration of hair’s pixels from the images. Moreover, we introduce a new combined loss function in the network’s training phase that combines the L_1 distance, the total variation loss, and a loss function based on the structural similarity index metric. For the evaluation of this method we have followed the same procedure previously described for the comparison of algorithms based on an larger database of simulated dermoscopic hair images. The results, both qualitatively and quantitatively, demonstrate the effectiveness of our model and how our loss function improves the restoration ability of the proposed model. A database has been created to evaluate the performance of the algorithms, accessible at http://dermaweb.uib.es/. The result of this work was compiled and published in an impact journal article in the first quartile [9].

D. Automatic detection of symmetry in dermoscopic images

Moreover, and given that the presence of asymmetries is a characteristic indicator of the presence of a malignant lesion,

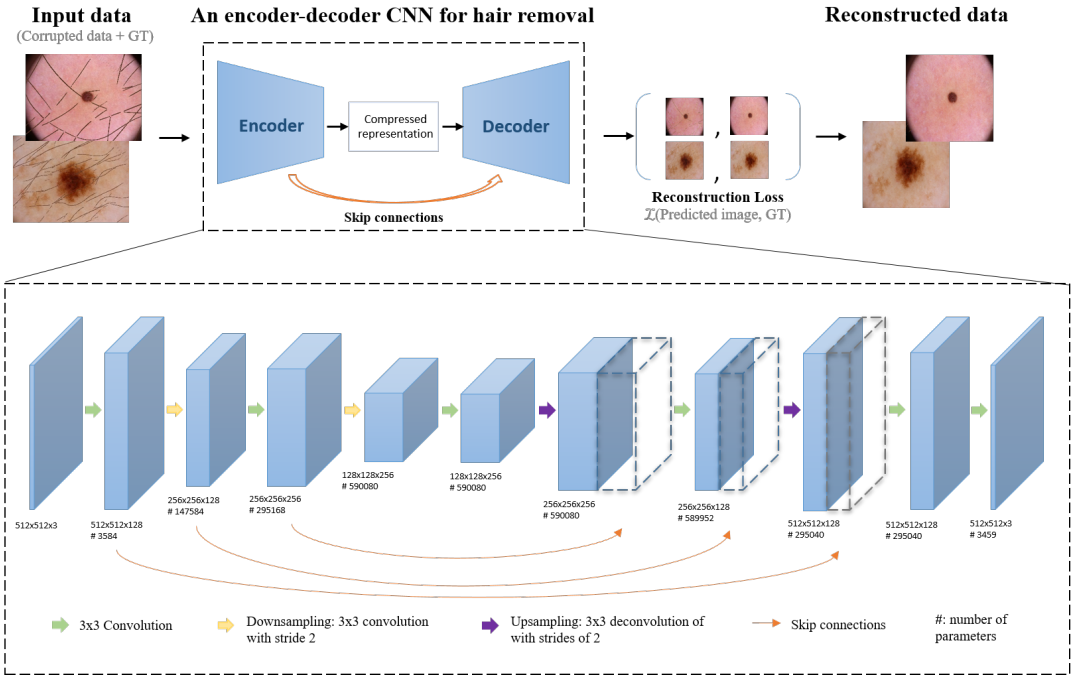
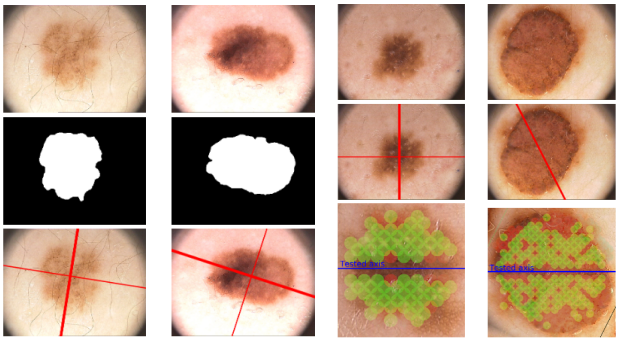


Fig. 3: Architecture of our proposed network. The pairs of reference hairless images (GT) and its corrupted (hair simulated) images are passed through the encoder to extract complex features. The decoder, connected to the encoder with skip-connections, reconstructs the image.

we decided to work on this aspect. A first work has been carried out in which we developed an automatic algorithm for detecting symmetries, as well as indicating the axes of symmetry. We tackle this task based on skin lesions' shape, color, texture, and their combination. To do so, we consider symmetry axes through the center of mass, random forests classifiers to aggregate across different orientations, and a purposely-built dataset to compare textures that are specific of dermoscopic imagery. In Figure 4, we can see some qualitative results. This work was accepted for presentation at the 18th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU 2020) [10]. However, we hope to be able to translate the method into a schematic using neural networks.



(a) (b)

Fig. 4: Examples of the automatic detection of symmetry in dermoscopic images based on their (a) shape and (b) texture.

E. How do different tasks influence each other's learning? Case study in dermoscopic images

Finally, we have started to work with multitasking learning with convolutional neural networks. What we intend is to provide a greater context to improve the result of related tasks, combining computational objectives with providing diagnostic tools that are interpretable by specialists. Multitask learning is arousing great interest for its ability to regularize models, or improve the generalization ability of the tasks that are learned simultaneously. Our first objective has been to design a multitask learning system, see Figure 5, with convolutional networks to simultaneously segment the skin lesion, segment the hairs of the lesion and perform the inpainting of these hairs in dermoscopic images. In addition, we study how different combinations of these tasks influence each other. The experiments are carried out on images from five publicly available datasets. We ascertain that optimizing a combined loss function while sharing hidden representations among the related tasks, may improve the ability to generalize when compared to each of the single-task contributions. This work in progress has been accepted at the IEEE 18th International Symposium on Biomedical Imaging (ISBI 2021) [11]. The next objective with this type of technique is to design a diagnostic tool based on the extracted characteristics, which tells the specialist not only that we are in the presence of a malignant skin lesion, but also why and on what such

decision is based: presence of regression zones, existence of asymmetries, etc.

F. System Validation

Regarding the third objective presented in Section I, the quantitative validation has already begun to be addressed since all the methods implemented in this project have been validated with expert information in a consistent and exhaustive way. This fact has allowed us to know if our methods are suitable for each specific application. However, the qualitative validation, as well as the ease of use and adoption by specialists, is expected to be addressed at the end of the thesis project.

G. Remaining milestones

Currently, we are working on the development of several algorithms for the analysis of different characteristics of skin lesions, based on deep neural networks. On the one hand, the first of them would focus on the segmentation of dermoscopic structures such as streaks, dots or pigment networks, as well as the classification of their presence as absent or present. On the other hand, the second algorithm would focus on classifying the symmetry of the lesion as asymmetric, symmetric with respect to 1 axis or symmetric with respect to 2 perpendicular axes.

Once we have completed these tasks, we will be able to integrate them into the multitasking system that we propose in this thesis project, and ascertain whether the interaction of different features from related-tasks allows us to gain a better assessment and understanding about the diagnosis of skin lesions.

III. Relevance

This project will contribute a better diagnosis of melanoma, which is an important, common and serious medical condition. Skin cancer is one of the most common cancers worldwide, with an increasing incidence rate in recent years. Melanoma is the most aggressive, metastatic and, therefore, the deadliest type of skin cancer. In Europe, cutaneous melanoma represents 1–2% of all malignant tumors [12]. Although melanoma is still incurable, its early diagnosis is of great importance, since it can prevent malignancy, and increase the survival rate and the effectiveness of the treatment. To detect melanoma at its earliest stage and to complement the clinical analysis, a screening device called dermoscopy is widely used by physicians. It is an in-vivo, non-invasive imaging technique used to amplify the lesion, allowing the visualization of structures, shapes and colors that are not discernible by a simple visual inspection. The use of dermoscopic images has shown to improve the diagnostic accuracy with respect to simple clinical observation: it can be up to 10–30% more accurate [13]. Taking into account this improvement, the diagnosis accuracy of expert dermatologists is estimated to be at 75–84% [14]. However, in this type of images, uncertainty and imprecision appear in all its amplitude: on the border of the lesion, in the characterization of some of its characteristics such as color, the presence or absence of some structures, etc. To minimize the number of unnecessary excisions that result from the visual inspection, which depends on the observers' experience and subjectivity [14], as well as, to provide an objective and reproducible diagnosis support tool, Computer Aided Diagnosis (CAD) systems have been developed.

In the literature, there are tons of works that have addressed the analysis of skin lesions on dermoscopic images within the field of computer vision, both with traditional and deep learning techniques. However, although the latter has proven to achieve higher performance for most applications within the medical field [15], there has always been a limiting factor for its practical use, as physicians may require explanations of the features involved in making decisions that make the model easy for them to interpret.

From a technical point of view, we explore the benefits of multitasking learning towards the explainability of deep learning models. This technique intends to simultaneously solve related tasks by learning a joint representation of some shared layers of the model, leveraging thus both their commonalities and differences [16]. Among its main advantages are the reduction of the computational time, an improvement in the model's robustness against overfitting, and a possible gain in prediction accuracy [17]. Until now, only a few works have relied on this interesting and powerful approach for the analysis of skin cancer on dermoscopic images. Most of them are focused on the simultaneous segmentation and classification of the lesion [18]–[22], whereas others segment the lesion attributes relying on classifying the ground truths as empty vs non-empty [20], or focus on classifying the categories that conform the 7-point checklist criteria to infer the diagnosis of the lesion [23].

We highlight the following aspects:

a) Societal impact: nowadays, the healthcare system is highly pressured, and the monitoring of patients has been more relevant than ever. So it is vitally important to focus on transferring knowledge to societal impact. This is the case of the implementation and development of CAD systems for the analysis of skin lesions, which can have a direct contribution to society, initially as an aiding tool for the physicians, and in the future helping specialists to reduce their workload by potentially offering these systems as a direct service to the population.

b) Innovation: this project may represent a significant step forward for the automatic and interpretable diagnosis of skin lesions. To the best of our knowledge, no previous works have addressed the automatic diagnosis by following a multitasking approach based on CNNs that aligns tasks like the lesion's symmetry classification, or the dermoscopic structures' segmentation, in addition to other well-known ones such as lesion segmentation and classification. Besides, we have created specific datasets of dermoscopic images annotated for the inpainting and symmetry classification tasks collected from publicly available datasets.

c) Technical relevance: we study the capacity and usefulness of neural networks as a technique capable of addressing

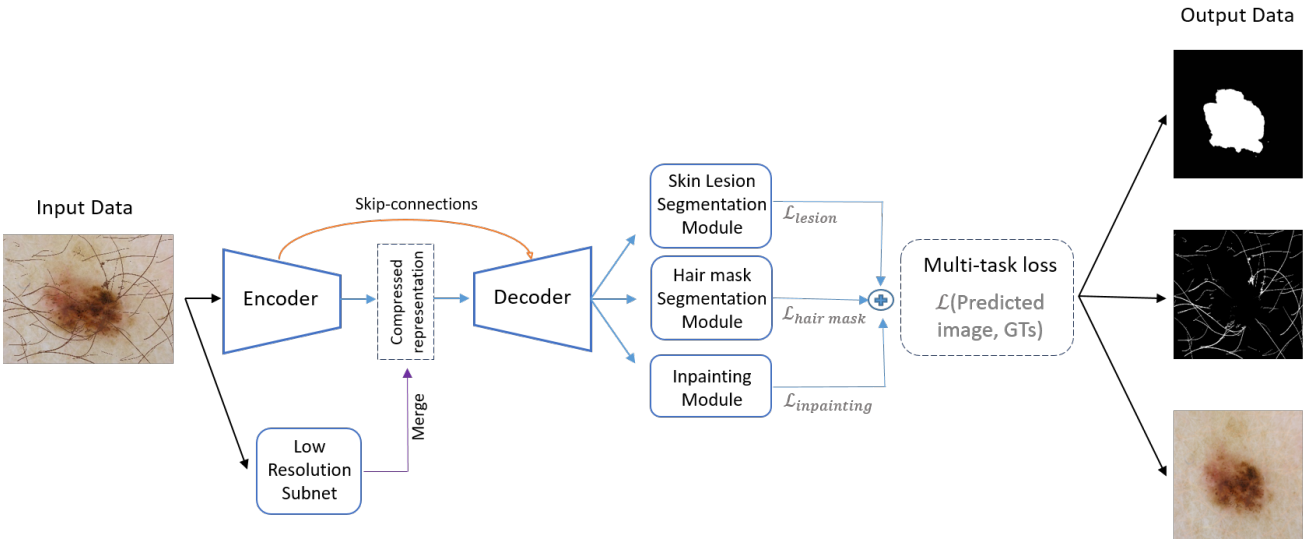


Fig. 5: Architecture of our multitasking network. The input of our model consists of a dermoscopic image and three ground-truth (GT) images, one for each task that the model intends to learn. The input flows in parallel through the low-resolution module and the encoder. Their two hidden representations are merged into one by concatenating their features, which are fed to the decoder, which also leverages skip connections to provide useful features for the output modules. Our model optimizes during training a weighted average of three loss functions.

a real problem such as the analysis of skin lesions in dermoscopic images. We expect the result of this research to improve and facilitate the interpretability of the final decision of the deep learning model by the experts when it comes to approve their validity. This self-explanation will be achieved by the agreement of the outputs of the different task that conforms the multitask learning approach.

References

[1] K. Korotkov and R. Garcia, "Computerized analysis of pigmented skin lesions: a review," *Artificial intelligence in medicine*, vol. 56, no. 2, pp. 69–90, 2012.

[2] S. Pathan, K. G. Prabhu, and P. Siddalingaswamy, "Techniques and algorithms for computer aided diagnosis of pigmented skin lesions—a review," *Biomedical Signal Processing and Control*, vol. 39, pp. 237–262, 2018.

[3] R. B. Oliveira, E. Mercedes Filho, Z. Ma, J. P. Papa, A. S. Pereira, and J. M. R. Tavares, "Computational methods for the image segmentation of pigmented skin lesions: a review," *Computer methods and programs in biomedicine*, vol. 131, pp. 127–141, 2016.

[4] L. Talavera-Martínez, P. Bibiloni, and M. González-Hidalgo, "Computational texture features of dermoscopic images and their link to the descriptive terminology: A survey," *Computer methods and programs in biomedicine*, vol. 182, p. 105049, 2019.

[5] ——, "Comparative study of dermoscopic hair removal methods," in *ECCOMAS Thematic Conference on Computational Vision and Medical Image Processing*. Springer, 2019, pp. 12–21.

[6] M. Attia, M. Hossny, H. Zhou, S. Nahavandi, H. Asadi, and A. Yazdabadi, "Realistic hair simulator for skin lesion images: A novel benchemarking tool," *Artificial Intelligence in Medicine*, vol. 108, p. 101933, 2020.

[7] H. Mirzaalian, "Hair Sim Software," http://www2.cs.sfu.ca/~hamarneh/software/hairsim/Welcome.html, accessed on 7th Mar 2019.

[8] F.-Y. Xie, S.-Y. Qin, Z.-G. Jiang, and R.-S. Meng, "PDE-based unsupervised repair of hair-occluded information in dermoscopy images of melanoma," *Computerized Medical Imaging and Graphics*, vol. 33, no. 4, pp. 275–282, 2009.

[9] L. Talavera-Martínez, P. Bibiloni, and M. González-Hidalgo, "Hair segmentation and removal in dermoscopic images using deep learning," *IEEE Access*, vol. 9, pp. 2694–2704, 2021.

[10] V. Toureau, P. Bibiloni, L. Talavera-Martínez, and M. González-Hidalgo, "Automatic detection of symmetry in dermoscopic images based on shape and texture," in *International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*. Springer, 2020, pp. 625–636.

[11] L. Talavera-Martínez, P. Bibiloni, and M. González-Hidalgo, "(in press) How do different tasks influence a multitasking environment? Case study in dermoscopic images," to be published at IEEE 18th International Symposium on Biomedical Imaging (ISBI 2021).

[12] Melanoma Molecular Map Project. http://www.mmmp.org/MMMP/welcome.mmmp. Accessed on 28th Feb 2019.

[13] J. Mayer *et al.*, "Systematic review of the diagnostic accuracy of dermatoscopy in detecting malignant melanoma," *Medical journal of Australia*, vol. 167, no. 4, pp. 206–210, 1997.

[14] G. Argenziano, H. P. Soyer, S. Chimenti, R. Talamini, R. Corona, F. Sera, M. Binder, L. Cerroni, G. De Rosa, G. Ferrara *et al.*, "Dermoscopy of pigmented skin lesions: results of a consensus meeting via the internet," *Journal of the American Academy of Dermatology*, vol. 48, no. 5, pp. 679–693, 2003.

[15] G. Litjens, T. Kooi, B. E. Bejnordi, A. Setio, F. Ciompi, M. Ghafoorian, J. Van Der Laak, B. Van Ginneken, and C. Sánchez, "A survey on deep learning in medical image analysis," *Medical image analysis*, vol. 42, pp. 60–88, 2017.

[16] R. Caruana, "Multitask learning," *Machine learning*, vol. 28, no. 1, pp. 41–75, 1997.

[17] S. Ruder, "An overview of multi-task learning in deep neural networks," *arXiv preprint:1706.05098*, 2017.

[18] X. Yang, Z. Zeng, S. Y. Yeo, C. Tan, H. L. Tey, and Y. Su, "A novel multi-task deep learning model for skin lesion segmentation and classification," *arXiv preprint:1703.01025*, 2017.

[19] S. Chen, Z. Wang, J. Shi, B. Liu, and N. Yu, "A multi-task framework with feature passing module for skin lesion classification and segmentation," in *IEEE 15th international symposium on biomedical imaging (ISBI)*, 2018, pp. 1126–1129.

[20] E. Z. Chen, X. Dong, X. Li, H. Jiang, R. Rong, and J. Wu, "Lesion attributes segmentation for melanoma detection with multi-task u-net," in *IEEE 16th International Symposium on Biomedical Imaging (ISBI)*, 2019, pp. 485–488.

[21] H. Liao and J. Luo, "A deep multi-task learning approach to skin lesion classification," *arXiv preprint:1812.03527*, 2018.

[22] L. Song, J. P. Lin, Z. J. Wang, and H. Wang, "An end-to-end multi-task deep learning framework for skin lesion analysis," *IEEE Journal of Biomedical and Health Informatics*, vol. 24, no. 10, pp. 2912–2921, 2020.

[23] J. Kawahara, S. Daneshvar, G. Argenziano, and G. Hamarneh, "Seven-point checklist and skin lesion classification using multitask multimodal neural nets," *IEEE journal of biomedical and health informatics*, vol. 23, no. 2, pp. 538–546, 2018.

Modelado Complejo de Información Lingüística en Problemas de Toma de Decisión en Grupo bajo Incertidumbre

Doctorando: Álvaro Labella Romero

Directores: Luis Martínez López y Rosa M. Rodríguez Domínguez

Grupo de investigación: Sistemas Inteligentes Basados en Análisis de Decisión Difusos (SINBAD^2)
Universidad de Jaén
Jaén, España
{alabella, martin, rmrodrig}@ujaen.es

Fecha de inicio: Septiembre de 2016

Palabras clave: toma de decisión lingüística, proceso de alcance de consenso, sistema de soporte a la decisión

I. Introducción

Los seres humanos deben de hacer frente, casi de manera constante, a situaciones de *Toma de Decisión*. Qué comer, qué ropa llevar o qué coche comprar, son sólo algunos ejemplos de este tipo de situaciones. Formalmente la TD se define como un proceso cognitivo en el que a través de diferentes procesos mentales y de razonamiento una persona selecciona, entre varias opciones o alternativas, la mejor como solución al problema [11]. Ya sea por la complejidad del problema o porque se exija en ciertos ámbitos de la sociedad, es muy común que la resolución de un problema de TD se ponga en manos de un conjunto de expertos, dando paso a los problemas de *Toma de Decisión en Grupo* (TDG) [5], [10].

La TDG implica inevitablemente considerar diferentes puntos de vista y, en consecuencia, los desacuerdos en el grupo [1]. Los esquemas de resolución clásicos obtenían la solución del problema mediante la agregación de las opiniones individuales de los expertos, ignorando esta cuestión [3]. Por lo tanto, era posible obtener una solución donde no todos los expertos estuvieran de acuerdo, sintiéndose fuera del proceso de decisión y poniendo en duda la fiabilidad del mismo. Un *Proceso de Alcance de Consenso* (PAC) se incluye en el esquema de resolución de un problema de TDG, antes de la selección de la mejor alternativa, con el objetivo de obtener una solución en consenso. En un PAC los expertos discuten y modifican sus opiniones iniciales con la idea de incrementar el nivel de acuerdo en el grupo. Los PAC se han convertido en una línea de investigación muy prolífica dentro de la TD, por lo que se han presentado numerosas propuestas de modelos de consenso en la literatura especializada [5], [6], [8]. Sin embargo, no existe ninguna medida objetiva que permita determinar qué modelo de consenso se adapta mejor a un problema de TDG [7].

Hoy en día, la sociedad demanda problemas de TDG y PAC cada vez más complejos. Esa complejidad genera falta de información e incertidumbre en los expertos, que no se sienten cómodos dando sus opiniones mediante valoraciones discretas que requieren demasiada precisión en un contexto con tanta incertidumbre. El *enfoque lingüístico difuso* [13] permite modelar dicha incertidumbre mediante información lingüística. De esta manera, los expertos se sienten más cómodos y seguros a la hora de expresar sus opiniones al usar expresiones lingüísticas cercanas a su forma de pensar. Bajo estas condiciones surge la *Toma de Decisión Lingüística* (TDL).

El modelado de incertidumbre mediante información lingüística conlleva manipular este tipo de información de alguna forma. La metodología de *Computación con Palabras* (CW) [12], [14] procesa palabras o frases dadas en un lenguaje natural o artificial en lugar de valores numéricos, imitando el razonamiento del ser humano. Por lo tanto, la idea principal de esta metodología es, partiendo de premisas lingüísticas, obtener resultados lingüísticos fáciles de entender. Numerosos enfoques de decisión aplican procesos de CW en sus modelos computacionales [9] aunque presentan varias limitaciones: (i) falta de expresividad en las expresiones lingüísticas, (ii) expresiones excesivamente complejas y (iii) falta de precisión e interpretabilidad en los resultados debido a la transformación de la información lingüística en otros formatos menos precisos y comprensibles.

La creciente complejidad en los problemas de TDG y sus PAC, unido a la aparición de incertidumbre, dificulta enormemente la labor de los expertos en el proceso de decisión. Los *Sistemas de Ayuda a la Decisión* (SAD) son herramientas

Esta tesis doctoral ha sido parcialmente financiada a través del proyecto de investigación TIN2015-66524-P otorgado por el Ministerio de Ciencia e Innovación del Gobierno de España.

software cuyo objetivo es proporcionar información adicional a los expertos sobre el problema de decisión y, de esta forma, ayudar a tomar la mejor decisión posible. A pesar de la evidente importancia de estos sistemas, existe una falta alarmante de propuestas de este tipo [4].

II. Hipótesis de Partida

Las principales limitaciones en los actuales modelos lingüísticos para la resolución de problemas de TDL, sus PAC y la falta de herramientas software para el tratamiento de dichos problemas, nos condujo al inicio de esta investigación a formular las siguientes hipótesis:

1. La definición de un nuevo y mejor marco metodológico a partir de modelos, metodologías y herramientas basadas en soft computing para el modelado difuso de la incertidumbre que, mediante modelos lingüísticos complejos para procesos de TDL y PAC, permitan superar distintos retos impuestos por las nuevas circunstancias y tendencias en las que han de desarrollarse dichos problemas y que actualmente no pueden resolverse.
2. La definición de una métrica para PAC que facilitará una mejor evaluación del funcionamiento de los distintos PAC actuales o de nuevas propuestas.
3. La aplicación de un nuevo marco metodológico en modelos de PAC y TDG. Además de su integración en un sistema software que producirá un importante avance en los PAC y TDG del mundo real al facilitar la resolución de problemas de forma automática y dar soporte a los expertos con herramientas comprensibles y adecuadas a los problemas.

III. Objetivos

Teniendo en cuenta las limitaciones previamente expuestas en los actuales modelos lingüísticos de TD y las hipótesis de partida, el propósito de esta tesis doctoral es investigar y definir modelos de TD y PAC lingüísticos que permitan superar las limitaciones comentadas anteriormente. En base a ésto, nos planteamos los siguientes objetivos:

1. Establecer un marco metodológico para el modelado y tratamiento de incertidumbre en TDG y sus PAC empleando expresiones lingüísticas complejas que permita modelar de forma apropiada las opiniones de los expertos y obtener resultados fácilmente interpretables y precisos.
2. Definir nuevos modelos lingüísticos de consenso para problemas de TDG bajo incertidumbre que superen las limitaciones de las propuestas existentes en la literatura especializada, mejorando la detección del disenso en el grupo y las recomendaciones de cambio sobre las opiniones de los expertos, y así incrementar el consenso entre expertos en el menor tiempo posible.
3. Elaborar métricas para procesos de consenso que establezcan referencias de funcionamiento en el alcance de consenso y de esta forma analizar y seleccionar el mejor PAC a aplicar en cada problema de TDG.
4. Investigar distintos problemas de TDG y PAC en el mundo real, identificando sus principales características y los retos que nos platean para así poder analizar y seleccionar el enfoque de resolución que proporciona la mejor solución posible.
5. Dar soporte a la TDG de los problemas anteriores mediante el desarrollo de SAD que ayuden a los expertos a manejar la creciente complejidad inherente en los problemas de TDG.

IV. Metodología y Plan de Trabajo

El análisis de las fuentes documentales relevantes se realizó vía el estudio de revistas de alto impacto en la temática, además de la revisión de bases de datos bibliográficas como Scopus o Web of Knowledge (WoK). Además, debido a los distintos objetivos que perseguimos, esta investigación comprende la obtención de resultados teóricos de distinto tipo y procesos de desarrollo software. A lo largo de la tesis doctoral utilizamos diferentes métodos de investigación dependiendo de las tareas que se llevan a cabo.

1. En la vertiente teórica empleamos una metodología de *Observación Activa*. Un estudio exhaustivo de modelos de decisión bajo incertidumbre basados en soft computing y sus procesos computacionales nos permitió analizar las ventajas e inconvenientes de dichas propuestas y proponer nuevas alternativas que superasen sus limitaciones.
2. En la vertiente de implementación, se incorporaron los nuevos resultados teóricos obtenidos a un SAD. Dicho sistema no se creó desde cero, ya que partimos de algunas herramientas enfocadas a la TD como el software registrado FLINTSTONES [2] y la herramienta AFRYCA [7] desarrollados por el grupo de investigación SINBAD^2. De esta forma, nuestro objetivo es ampliar la funcionalidad de dichas herramientas para construir un SAD más completo.

Para alcanzar los objetivos anteriormente mencionados usando esta metodología se definieron las siguientes tareas como parte del plan de trabajo:

- **Tarea 1**: Recopilación y estudio bibliográfico tanto de los conceptos relacionados con los objetivos de la línea teórica como de la línea práctica.
  - Hito 1.1: Estudio del estado del arte.
  - Entrega 1.1: Documento técnico de la revisión.
- **Tarea 2**: Marco metodológico para el modelado y tratamiento de incertidumbre en TDG y en sus PAC orientado al empleo de expresiones lingüísticas complejas.
  - Hito 2.1: Definición de un marco metodológico para TDG bajo incertidumbre.
  - Hito 2.2: Definición de PAC para TDG.
  - Hito 2.3: Definición de métricas para PAC.
  - Entrega 2.1: Documento técnico sobre los procesos y herramientas de estudio a realizar en la investigación.
  - Entrega 2.2: Documento técnico sobre los PAC definidos.

  - Entrega 2.3: Documento técnico sobre las métricas definidas.
- **Tarea 3**: Implementación de los modelos lingüísticos de TDG en un SAD en grupo.
  - Hito 3.1: Diseño del prototipo.
  - Hito 3.2: Implementación del prototipo y su funcionalidad.
  - Hito 3.3: Proceso de integración y pruebas.
  - Entrega 3.1: Prototipo implementado.
  - Entrega 3.2: Documento con pruebas y manual de usuario.
- **Tarea 4**: Aplicación del prototipo implementado en la Tarea 3 en diversos problemas de TDG del mundo real.
  - Hito 4.1: Análisis y estudio de casos reales que pueden ser aplicados al prototipo diseñado.
  - Entrega 4.1: Documento técnico sobre problemas de TD reales.
  - Entrega 4.2: Diseño de los casos reales estudiados para que puedan ser analizados mediante el prototipo.
- **Tarea 5**: Redacción de la memoria de Tesis Doctoral, Informe y Conclusiones.
  - Hito 5.1: Redacción de memoria de tesis doctoral.
  - Entrega 5.1: Memoria de tesis doctoral.
  - Entrega 5.2: Artículos científicos.

V. Relevancia

La investigación llevada a cabo en esta tesis doctoral ha dado como resultado, hasta este momento, un total de diez publicaciones en revistas internacionales indexadas con un alto índice de impacto. A nivel científico, se han presentado ocho publicaciones:

1. Á. Labella, R. M. Rodríguez, A. A. Alzahrani and L. Martínez. A consensus model for extended comparative linguistic expressions with symbolic translation. Mathematics, 8(12):2198, 2020.
2. Á. Labella, R. M. Rodríguez and L. Martínez. Computing with comparative linguistic expressions and symbolic translation for decision making: ELICIT information. IEEE Transactions on Fuzzy Systems, 28(10):2510–2522, 2020.
3. Á. Labella, H. Liu, R. M. Rodríguez and L. Martinez. A cost consensus metric for consensus reaching processes based on a comprehensive minimum cost model. European Journal of Operational Research, 281(2):316–331, 2020.
4. Á. Labella, R. M. Rodríguez and L. Martínez. A consensus reaching process dealing with comparative linguistic expressions for group decision making: A fuzzy approach. Journal of Intelligent & Fuzzy Systems, 38(1):735–748, 2020.
5. B. Dutta, Á. Labella, R. M. Rodríguez and L. Martínez. Aggregating interrelated attributes in multi-attribute decision-making with elicit information based on bonferroni mean and its variants. International Journal of Computational Intelligence Systems, 12:1179 – 1196, 2019.
6. Á. Labella, Y. Liu, R. M. Rodríguez and L. Martínez. Analyzing the performance of classical consensus models in large scale group decision making: A comparative study. Applied Soft Computing, 67:677–690, 2018.
7. R. M. Rodríguez, Á. Labella, G. De Tré and L. Martínez. A large scale consensus reaching process managing group hesitation. Knowledge-Based Systems, 2018.
8. R. M. Rodríguez, Á. Labella and L. Martínez. An overview on fuzzy modelling of complex linguistic preferences in decision making. International Journal of Computational Intelligence Systems, 9:81–94, 2016.

Los avances en la vertiente práctica de la tesis doctoral han dado lugar al desarrollo de dos SAD presentados en los siguientes artículos:

1. Á. Labella, K. Koasidis, A. Nikas, A. Arsenopoulos, and H. Doukas. APOLLO: A fuzzy multi-criteria group decision-making tool in support of climate policy. International Journal of Computational Intelligence Systems, 13(1):1539–1553, 2020.
2. Á. Labella, F.J. Estrella, and L. Martínez. AFRYCA 2.0: an improved analysis framework for consensus reaching processes. Progress in Artificial Intelligence, 6:181–194, 2017.

Actualmente, seguimos trabajando en dos propuestas. El primer trabajo consiste en la definición de una nueva medida de cohesión que nos permita medir el grado de cercanía de las opiniones de los expertos en problemas de TDG bajo incertidumbre. Esta medida será usada para ponderar de manera más apropiada sub-grupos de expertos obtenidos mediante técnicas de clustering en problemas de TDG a gran escala. La segunda propuesta consiste en extender uno de los trabajos ya publicados donde presentamos los modelos de consenso de mínimo coste integrales. Estos modelos permiten obtener soluciones óptimas para un PAC y además, garantizan un nivel de consenso deseado en el grupo de expertos. Sin embargo, la naturaleza matemática de estos modelos basada en programación no lineal los hace prácticamente inutilizables para resolver problemas de TDG con un gran número de expertos involucrados en el proceso de decisión. Por esta razón, nuestra idea es proponer nuevos modelos de consenso de mínimo coste integrales lineales que sean capaces de manejar problemas con cientos de expertos y que, además, garanticen la consistencia en sus opiniones.

Referencias

[1] P. Eklund, A. Rusinowska, and H. de Swart. A consensus model of political decision-making. *Annals of Operations Research*, 158(1):5–20, 2008.

[2] Francisco J Estrella, Macarena Espinilla, Francisco Herrera, and L Martínez. Flintstones: A fuzzy linguistic decision tools enhancement suite based on the 2-tuple linguistic model and extensions. *Information Sciences*, 280:152–170, 2014.

[3] F. Herrera, E. Herrera-Viedma, and J. L. Verdegay. A sequential selection process in group decision making with a linguistic assessment approach. *Information Sciences*, 85(4):223–239, 1995.

[4] A. Ishizaka and A. Labib. Analytic hierarchy process and expert choice: Benefits and limitations. *Or Insight*, 22(4):201–220, 2009.

[5] Á. Labella, Y. Liu, R. M. Rodríguez, and L. Martínez. Analyzing the performance of classical consensus models in large scale group decision making: A comparative study. *Applied Soft Computing*, 67:677–690, 2018.

[6] C.C Li, Y. Dong, and F. Herrera. A consensus model for large-scale linguistic group decision making with a feedback recommendation based on clustered personalized individual semantics and opposing consensus groups. *IEEE Transactions on Fuzzy Systems*, 27(2):221–233, 2019.

[7] I. Palomares, F. J. Estrella, L. Martínez, and F. Herrera. Consensus under a fuzzy context: Taxonomy, analysis framework afryca and experimental case of study. *Information Fusion*, 20:252–271, 2014.

[8] R. M. Rodríguez, Á. Labella, G. De Tré, and L. Martínez. A large scale consensus reaching process managing group hesitation. *Knowledge-Based Systems*, 2018.

[9] R.M. Rodríguez, A. Labella, and L. Martínez. An overview on fuzzy modelling of complex linguistic preferences in decision making. *International Journal of Computational Intelligence Systems*, 9:81–94, 2016.

[10] O. Sohaib, M. Naderpour, W. Hussain, and L. Martinez. Cloud computing model selection for e-commerce enterprises using a new 2-tuple fuzzy linguistic decision-making method. *Computers & Industrial Engineering*, 132:47–58, 2019.

[11] E. Ward. The theory of decision making. *Psychological bulletin*, 51(4):380, 1954.

[12] R. R. Yager. Approximate reasoning as a basis for computing with words. In *Computing with words and information/intelligent systems 2: applications*, pages 50–77. Physica Verlag, 1999.

[13] L. A. Zadeh. The concept of a linguistic variable and its application to approximate reasoning, part i, ii, iii. *Information Sciences*, 8,8,9:199 – 249, 301–357, 43–80, 1975.

[14] L. A. Zadeh. Fuzzy logic = computing with words. *IEEE Transactions on Fuzzy Systems*, 4(2):103–111, 1996.

Nuevas aproximaciones de aprendizaje por transferencia en sistemas de recomendación

Eva Blanco-Mallo*, Beatriz Remeseiro†§ and Verónica Bolón-Canedo*§
* *Departamento de Computación, Universidade da Coruña, CITIC*
Campus de Elviña, 15071 A Coruña, España
eva.blanco@udc.es, vbolon@udc.es
† *Departamento de Informática, Universidad de Oviedo*
Campus de Gijón s/n, 33203 Gijón, España
bremeseiro@uniovi.es

***Abstract*—Hoy en día, el éxito del aprendizaje automático es indudable. Dado que su principal objetivo es establecer patrones de comportamiento a partir de conjuntos de datos, su eficacia radica en los mismos, tanto en cantidad como en calidad. Recolectar y etiquetar grandes volúmenes de datos es una tarea ardua y costosa, inabordable en multitud de escenarios. El aprendizaje por transferencia es un área de investigación dentro del aprendizaje automático centrada en la reutilización de conocimiento entre sistemas. Esto abre un amplio abanico de posibilidades, ya que permite afrontar aquellas tareas en las que la disponibilidad de los datos sea un problema. Nuestra investigación se centra en explorar nuevas aproximaciones y aplicaciones del aprendizaje por transferencia en el contexto de los sistemas de recomendación. En este campo es común contar con bases de datos desbalanceadas e información escasa o nula en gran parte de los usuarios. Nuestro objetivo es aplicar aprendizaje por transferencia para tratar de solventar estos obstáculos y mejorar la calidad de la recomendación.**

***Index Terms*—transferencia de conocimiento, aprendizaje automático, sistemas de recomendación, extracción de características**

I. Introducción

El aprendizaje automático ha demostrado ser capaz de resolver exitosamente multitud de problemas del mundo real en prácticamente todas las áreas de conocimiento. Su fuerte reside en su habilidad para extraer patrones de comportamiento en conjuntos de datos, buscando la función que mejor estime la distribución de los mismos en función de una salida deseada. Con esta función, será posible predecir futuras salidas en base a nuevos datos. Debido a su fuerte componente estadístico, cuanto mayor sea el conjunto de datos manejados por un sistema, mejor será la generalización de comportamientos e inferencias, y mejor será la predicción futura ante nuevos datos. Por tanto, encontrar el conjunto de entrenamiento que mejor represente el dominio que queremos aproximar es un problema crucial en el aprendizaje automático. Sin embargo, recopilar tantos datos como requiere el aprendizaje automático, y sobre todo etiquetarlos, es una tarea que consume una gran cantidad de tiempo y recursos, resultando a veces una tarea completamente inviable. Ya sea porque los datos son poco frecuentes, peculiares o simplemente inaccesibles.

La intuición detrás del aprendizaje por transferencia (*transfer learning*) es que el conocimiento aprendido en la resolución de una tarea previa podría ayudar en la resolución de nuevas tareas, ya sea mejorando el rendimiento alcanzado o reduciendo la velocidad necesaria para poder aprender. Por ejemplo, partiendo de las mismas capacidades como base, un guitarrista profesional indudablemente aprendería a tocar el piano antes que una persona que nunca tocó un instrumento musical. Concretamente en aprendizaje automático, partiendo de una tarea T que tenemos que resolver en un dominio D determinado, el conocimiento alcanzado en el dominio fuente (*source*) D_S para la resolución de la tarea T_S podría ayudar en el aprendizaje del dominio objetivo (*target*) D_T enfocado a la resolución de una nueva tarea T_T. Por tanto, con la llegada del aprendizaje por transferencia, surge un nuevo horizonte dentro del aprendizaje automático, que permitiría abordar los problemas anteriormente mencionados.

Actualmente nos centramos en explorar las ventajas de aplicar *transfer learning* en el contexto de los sistemas de recomendación (SR). Un SR es un sistema de filtrado de información sobre un determinado tipo de elementos, cuyo objetivo es encontrar recomendaciones para un usuario determinado. Este tipo de sistemas trabajan con datos diádicos, que se refieren a un dominio con dos conjuntos finitos de objetos en los que las observaciones se realizan por díadas, es decir, pares con un elemento de cada conjunto. Tomando como ejemplo los datos disponibles en TripAdvisor, una de las mayores plataformas de viajes del mundo, los pares podrían estar formados por usuarios y restaurantes, y el objetivo principal del SR sería predecir la relación entre ellos. Los SR están cada vez más presentes en nuestra vida cotidiana, especialmente desde la llegada del Big Data, que permite almacenar todo tipo de información sobre las preferencias de los usuarios. Los SR personalizados se aplican con éxito en plataformas como Netflix, Amazon o YouTube. Sin embargo,

Este trabajo ha sido financiado por el Plan Estatal de Investigación Científica y Técnica y de Innovación del Gobierno de España (Subvención PID2019-109238GB, subproyectos C21 and C22), y por la Xunta de Galicia (Subvención ED431C 2018/34) con fondos FEDER de la Unión Europea. El CITIC, como Centro de Investigación acreditado por el Sistema Universitario de Galicia, está financiado por la "Consellería de Cultura, Educación e Universidades de la Xunta de Galicia", apoyada en un 80% a través de Fondos FEDER, Programa Operativo FEDER Galicia 2014-2020, y el 20% restante por la "Secretaría Xeral de Universidades" (Subvención ED431G 2019/01).

§These authors jointly supervised this work.

plataformas gastronómicas como TripAdvisor, donde podemos encontrar millones de imágenes etiquetadas con los gustos de los usuarios, se limitan a mostrar los restaurantes por orden de popularidad. Por tanto, podemos hacernos una idea del impacto económico de utilizar estos datos en un sistema de recomendación personalizado.

A. Fundamentos del aprendizaje por transferencia

Podemos encontrar diferencias de varios tipos entre tareas y dominios a la hora de definir un problema concreto de transferencia de conocimiento, como veremos a continuación. Un dominio D está definido por un espacio de características $\mathcal{X}$ y una distribución de probabilidad $P(X)$, donde cada elemento de $X = \{x_1, ..., x_n\}, \forall x \in \mathcal{X}$, está definido por un vector de características. A su vez, una tarea T está definida por un espacio de etiquetas $\mathcal{Y}$ y una función de predicción $f(.)$. Cuando queremos entrenar un sistema para resolver la tarea $T = \{\mathcal{Y}, f(.)\}$ en el dominio $D = \{\mathcal{X}, P(X)\}$ utilizando aprendizaje automático, tenemos que inferir la función predictiva $f(.)$ que mejor estime la distribución de probabilidad $P(X)$ asociada a Y (una muestra del espacio de etiquetas $\mathcal{Y}$) a partir de los datos de entrada X (una muestra del espacio de búsqueda $\mathcal{X}$). Si tenemos un sistema H que ha sido entrenado para resolver la tarea T_S en el dominio D_S, podríamos utilizar ese conocimiento para resolver una nueva tarea T_T en el dominio D_T, y según la definición formal anteriormente aportada, podríamos encontrar las siguientes diferencias:

1) Los **dominios son diferentes**, tal que $\boldsymbol{D_S \neq D_T}$. Esta condición puede implicar que los espacios de búsqueda son diferentes o que las distribuciones de probabilidad lo son, aunque las tareas fuente y objetivo son iguales.
   - $\mathcal{X}_\mathcal{S} \neq \mathcal{X}_\mathcal{T}$. En este caso, aunque la tarea y la distribución de probabilidad sean las mismas, el espacio de características cambia y, por tanto, los datos de entrada. Por ejemplo, queremos que un sistema que aprendió a reconocer textos en un idioma determinado pueda aprender a reconocerlos en un nuevo idioma.
   - $P(X_S) \neq P(X_T)$. El espacio de características entre ambas tareas se mantiene, lo que varía es la distribución de probabilidad de las mismas. Por ejemplo, queremos aprender a clasificar documentos de carácter legal a partir de un sistema que clasifica documentos de carácter administrativo.
2) Las **tareas son diferentes**, tal que $\boldsymbol{T_S \neq T_T}$. Pudiendo diferir en el espacio de etiquetas o en la función de predicción.
   - $\mathcal{Y}_\mathcal{S} \neq \mathcal{Y}_\mathcal{T}$. Al cambiar el espacio de etiquetas, bajo las mismas circunstancias, la salida tiene significados diferentes en ambas tareas. Por ejemplo, si tenemos un sistema binario que detecta pacientes con diabetes de pacientes sin ella y ahora queremos que la salida clasifique a los pacientes en función de su tipo de diabetes.
   - $f_S(.) \neq f_T(.)$. La función $f(.)$ puede ser usada para predecir la correspondiente etiqueta $y_i \in \mathcal{Y}$ asociada a la muestra $x_i \in \mathcal{X}$, tal que $f(x_i) = y_i$. Desde un punto de vista probabilístico, podemos describir la función de predicción $f(X)$ como $P(X|Y)$. Podemos usar el caso de la diabetes anterior para ejemplificar esta diferencia. Imaginemos que un sistema binario que detecta pacientes con diabetes de tipo 1 ahora debe detectar pacientes con diabetes de tipo 2.

Así mismo, podrían darse varias diferencias simultáneamente entre dominios y tareas, siendo también un tema importante a considerar la disponibilidad de datos etiquetados y no etiquetados en ambas fuentes. También es interesante destacar la transferencia de conocimiento negativa (*negative transfer learning*), que ocurre cuando el rendimiento en la tarea objetivo se ve degradada al utilizar información de un dominio fuente junto con la de su propio dominio. Concretamente, el *negative transfer* se centra en cuantificar el nivel relación entre dominios para decidir cuándo es recomendable llevar a cabo un procedimiento de *transfer learning* y cuándo no.

En la literatura podemos distinguir entre cuatro enfoques principales:

- *Instance-based transfer learning*: basado mayormente en el ajuste de pesos con el objetivo de adaptar la distribución de probabilidad del dominio fuente al dominio objetivo.
- *Feature-based transfer learning*: se centra en aprender una representación de características que pueda resultar útil para el dominio objetivo haciendo uso del dominio fuente, ya sea transformando las características del dominio fuente a través del ajuste de pesos para que coincida con el del objetivo o descubriendo estructuras relacionadas entre dominios para encontrar un espacio de características común.
- *Parameter-based transfer learning*: trata de transferir conocimiento compartiendo parámetros entre modelos, ya sea entre modelos entrenados para resolver independientemente las tareas fuente y objetivo o creando múltiples modelos a través diferentes fuentes, que se ensamblan posteriormente siguiendo un ajuste de pesos para encontrar la solución óptima que mejore el rendimiento en el modelo objetivo.
- *Relational-based transfer learning*: asume que la tarea objetivo y la tarea fuente comparten una serie de relaciones que pueden ser utilizadas para transferir conocimiento.

B. El camino hasta ahora

En trabajos anteriores [2], [3] propusimos un sistema de recomendación basado en imágenes utilizando las opiniones de los usuarios de TripAdvisor, definiendo el problema como una tarea de clasificación en la que tenemos tríadas con una de dos etiquetas:

$$(u, r, i) \rightsquigarrow 0|1, \tag{1}$$

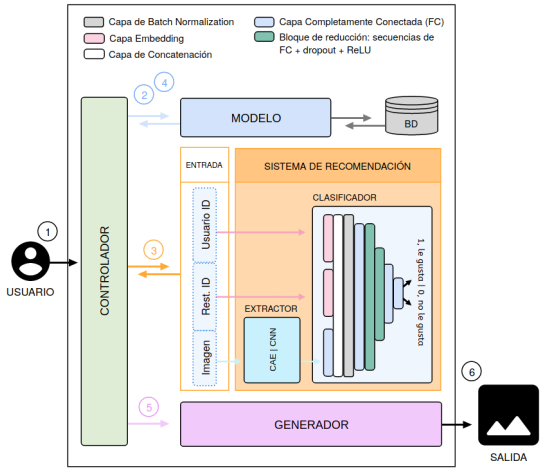


Fig. 1. Arquitectura del método propuesto para ofrecer recomendaciones personalizadas basadas en imágenes.



Fig. 2. Salida del sistema propuesto para un usuario determinado.

donde u es un usuario que visitó el restaurante r y sacó la imagen i. En cuanto a las dos etiquetas, 0 significa que al usuario u no le gusta el restaurante r, mientras que 1 es lo contrario. Con el objetivo de resolver esta tarea, propusimos el modelo representado en la Figura 1: una red que aprende sobre tríadas de usuarios, restaurantes y fotos (u, r, i) para proporcionar una recomendación personalizada basada en imágenes. La salida proporcionada por el sistema se puede ver en la Figura 2.

Los datos utilizados para la experimentación se recopilaron en 2018–2019 a partir de reseñas de restaurantes de TripAdvisor en tres ciudades de diferente tamaño [6]: Santiago de Compostela (SGC), Barcelona (BCN) y Nueva York (NYC). El conjunto de datos original se dividió para obtener los conjuntos de entrenamiento y test, con la idea de asegurar que todos los usuarios y restaurantes evaluados en el conjunto de test estén también en el conjunto de entrenamiento. Usuarios y restaurantes son representados mediante una codificación one-hot y mapeados a embeddings de 512 dimensiones, mientras que las fotos son codificadas utilizando un autoencoder convolucional (CAE). El CAE, basado en el propuesto por Chollet [4], fue entrenado con todas las imágenes disponibles y utilizado posteriormente para codificar las imágenes de entrada. En cuanto al aprendizaje por transferencia, se contrastó la eficacia del CAE como extractor de características con una red neuronal convolucional (CNN), considerada un referente en la clasificación supervisada de imágenes. En concreto, probamos el enfoque de aprendizaje de transferencia basado en parámetros utilizando ReNet50 [7] con pesos pre-entrenados en ImageNet [5], con y sin ajuste de parámetros.

Precisión y F1-score son dos de las métricas más populares en SR. Sin embargo, la precisión no funciona bien en conjuntos de datos desbalanceados, como es nuestro caso, y el F1-score da más importancia a la clase positiva, por lo que los problemas con la clase negativa pueden pasar desapercibidos. En un sistema de recomendación no sólo es importante recomendar aquellos artículos que le gustan al usuario, sino también no recomendar aquellos artículos que no le gustan. Por lo tanto, buscamos un sistema con valores altos de sensibilidad y especificidad. Para ello, propusimos utilizar una medida equilibrada, balanced score (B-score), una variación del F1-score que calcula la media armónica de la sensibilidad y la especificidad:

$$\text{B-score} = 2 * \frac{\text{sensibilidad} * \text{especificidad}}{\text{sensibilidad} + \text{especificidad}} \quad (2)$$

Si nos basamos en métricas como la precisión o el F1-score para seleccionar el mejor modelo, podríamos elegir el que mejor detecte los elementos que le gustan al usuario, aunque el rendimiento sea muy bajo a la hora de clasificar los elementos que no le gustan (por ejemplo, 0.99 de sensibilidad y 0.40 de especificidad: F1-score = 0.94, B-score = 0.56), en lugar de uno con predicciones más equilibradas y deseables (por ejemplo, 0.85 de sensibilidad y 0.70 de especificidad: F1-score = 0.89, B-score = 0.76). Utilizando el B-score, el modelo elegido será el que tenga el mejor equilibrio entre sensibilidad y especificidad.

A la vista de los resultados, el mejor B-score fue el alcanzado por el CAE, el extractor que solo hace uso del conocimiento propio del dominio objetivo al aplicar aprendizaje por transferencia. La diferencia se hace más notable al examinar los resultados de especificidad. Como era de esperar, la peor opción fue utilizar la ResNet50 sin ajuste de parámetros porque, aunque fue entrenada con una base de datos mucho mayor (ImageNet), el propósito del entrenamiento era resolver una tarea de clasificación de imágenes, que no está relacionada con el problema que nos ocupa; por lo tanto, no se tiene en cuenta el contexto que involucra a usuarios y restaurantes. Con ajuste de parámetros, la CNN consigue mejores resultados, sin embargo, presenta un rendimiento menor que con el CAE. Por tanto, el uso del aprendizaje de transferencia no fue tan útil como se pretendía y se obtienen mejores resultados utilizando sólo el conocimiento específico del dominio objetivo.

C. Objetivos

Nuestros objetivos se centran en la búsqueda de nuevos métodos o aplicaciones del aprendizaje por transferencia, principalmente entre sistemas de recomendación. La intención

es profundizar en el tema en general y abordar diferentes cuestiones que surgieron en trabajos de investigación anteriores, concretamente abordando los siguientes puntos:

1) El desbalanceo de clases es muy común en los problemas de recomendación, tanto en general en el conjunto de datos, donde la gran mayoría de reseñas es positiva, como a nivel de usuario, siendo frecuentes los casos de usuarios con reseñas de una sola clase. Por ello, nos centraremos en buscar formas de reutilizar el conocimiento obtenido de los usuarios con perfiles más completos (es decir, con valoraciones positivas y negativas en su historial) en aquellos usuarios con poca información disponible. La idea es que, partiendo del conocimiento extraído de usuarios completos, podamos obtener una base que nos permita abordar la recomendación no sólo ante perfiles incompletos, sino también ante la ausencia de información de nuevos usuarios.
2) Los problemas de recomendación y aprendizaje de transferencia tienen un aspecto en común: ambos hacen uso de medidas de distancia. En concreto, en el aprendizaje de transferencia es necesario evaluar la diferencia entre las distribuciones de los conjuntos fuente y objetivo y los sistemas de recomendación también hacen uso de medidas de similitud para comparar los perfiles de los usuarios. Por ello, actualmente estamos estudiando el uso de estas medidas en el contexto del aprendizaje automático, analizando sus principales propiedades y profundizando en las métricas más utilizadas.
3) Seguiremos explorando el *parameter-based transfer learning*. A diferencia del caso presentado en la Sección I-B, haremos uso de información relacionada con el dominio objetivo únicamente, con el fin de reutilizar la información aprendida a partir de los usuarios de una ciudad en los usuarios de una ciudad diferente.
4) También queremos abordar la recomendación de otro tipo de productos. El propósito es ver si las características visuales que definen los gustos de los usuarios, en cuanto a valoraciones gastronómicas se refiere, son útiles para abordar otro tipo de recomendaciones también basadas en imágenes.
5) Además de utilizar las imágenes presentes en las reseñas de los usuarios, también trataremos sus comentarios de texto. De esta manera se pueden analizar los efectos de las técnicas de aprendizaje de transferencia aplicadas a diferentes tipos de datos. El objetivo principal es utilizar todo el conocimiento disponible de cada usuario, independientemente del tipo de dato con el que esté representado, pudiendo aplicar las mejores técnicas de transferencia en cada caso para obtener resultados óptimos.

Como se puede observar, nuestros objetivos están relacionados fundamentalmente con aquellos casos de transferencia de conocimiento en los que las tareas fuente y objetivo están relacionadas pero existen diferencias entre ambos dominios, ya sea en el espacio de características o en la distribución de las mismas. Un subcampo del aprendizaje por transferencia centrado específicamente en estos casos es el denominado *domain adaptation* (DA), cuyo objetivo es alterar el dominio fuente para acercar su distribución a la del dominio objetivo.

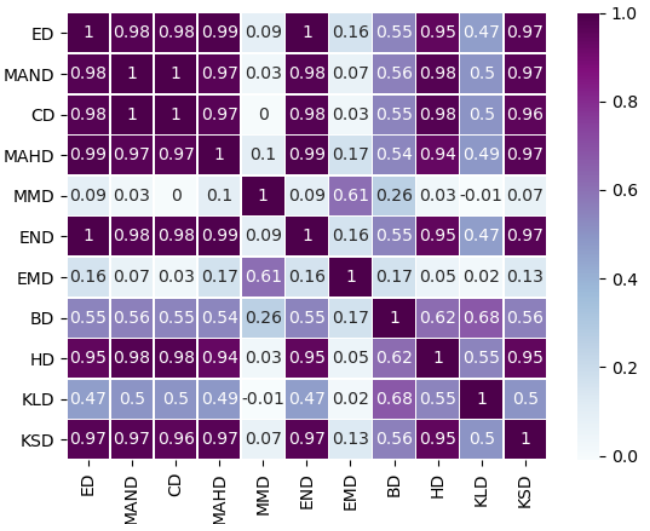


Fig. 3. Correlación entre los resultados de las diferentes medidas de distancia analizadas ante los mismos datos de entrada. Abreviaturas utilizadas: ED: distancia Euclídea, MAND: distancia de Manhattan, CD: distancia del coseno, MAHD: distancia de Mahalanobis, MMD: maximum mean discrepancy, END: distancia energy, EMD: earth mover's distance, BD: distancia de Bhattacharyya, HD: distancia de Hellinger, KLD: distancia de Kullback-Leibler, KSD: distancia de Kolmogorov-Smirnov.

II. Metodología y plan de trabajo

Para abordar con éxito cualquier investigación, es importante conocer bien las bases. Muchos sistemas de recomendación y aprendizaje por transferencia tienen al menos un denominador común: se basan en el uso de medidas de distancia. Por ello, hemos analizado las principales contribuciones que tratan sobre este tema y seleccionado las medidas más utilizadas en los últimos años [1], discutiendo sus principales propiedades y aplicaciones. En total estudiamos el rendimiento de 11 medidas de distancia, entre ellas algunas altamente conocidas, como las distancias Euclídea, Manhattan o Hellinger, y otras más recientes, como maximum mean discrepancy o earth mover's distance. Además, se han realizado varios experimentos para analizar su similitud (ver Figura 3) y contrastar su rendimiento en tareas comunes de aprendizaje automático, como clasificación o clustering.

Continuando con el trabajo de partida expuesto en la Sección I-B, nuestra intención es seguir explorando el aprendizaje por transferencia basado en parámetros, *parameter-based transfer learning*, para mejorar el proceso de extracción de características. Estudiaremos en profundidad cómo utilizar la información aprendida por el autoencoder en una ciudad representativa para mejorar la recomendación en el resto de ciudades. Por lo tanto, a diferencia del enfoque anterior, siempre utilizaremos datos pertenecientes al dominio objetivo, pero buscando la mejor manera de hacerlo. Un problema importante

con el que nos enfrentamos al trabajar con arquitecturas tan profundas y tipos de datos complejos es el tiempo de entrenamiento que requieren. Una disminución del tiempo de entrenamiento supondría una reducción de costes considerable y permitiría invertir recursos en otras tareas, como llevar a cabo más pruebas o ajuste de hiperparámetros, que son esenciales en este tipo de modelos. Además del tiempo de entrenamiento, otro reto importante al que nos enfrentamos es la falta de datos representativos sobre las preferencias de los usuarios, un problema típico en las aplicaciones de reseñas. La gran mayoría de los perfiles tienen una única valoración, mayoritariamente positiva. A diferencia de otras plataformas, en las que podíamos hacer uso de otro tipo de información, como los historiales de compra, en este caso no disponemos de un registro de todos los restaurantes visitados por el usuario, le hayan gustado o no. Con una información tan limitada, la tarea de ofrecer una recomendación personalizada adecuada es muy compleja. Para resolver este problema, estudiaremos diferentes enfoques orientados a aprender una representación de características comunes a partir de usuarios con perfiles completos que puedan ser útiles cuando el sistema se enfrenta a usuarios similares con información limitada. La intención es que este conocimiento pueda utilizarse también con nuevos usuarios.

También examinaremos diferentes alternativas para abordar el problema de recomendación, en concreto, identificando primero cuáles son los factores que más llaman la atención de los usuarios a la hora de publicar una reseña, ya sea positiva o negativa. Para ello, analizaremos los comentarios de texto de los usuarios relacionados con puntos de interés (POI), lugares concretos que podrían ser útiles o interesantes. TripAdvisor también dispone de una gran cantidad de información sobre puntos turísticos o emblemáticos de las ciudades, así como de restaurantes, por lo que seguiremos utilizando los datos de esta plataforma. Suponemos que si hay ciertos factores clave que influyen en la percepción que los usuarios tienen de un lugar, probablemente se reflejen en sus comentarios. Y si es un factor relevante, se repetirá con cierta frecuencia. El análisis de sentimientos (*sentimental analysis*) se utilizará para analizar la polaridad de los textos, extrayendo los temas más relevantes dentro de cada clase, positiva y negativa. Este análisis se realizará tanto a nivel de usuario como de POI, con el objetivo final de integrar la información extraída en un sistema de recomendación. La clave es encontrar los POIs con más factores relevantes positivos en común con un usuario objetivo y menos factores relevantes negativos. Al igual que en el caso de uso comentado hasta ahora (Sección I-B), trataremos de resolver el problema relacionado con la falta de información y el desbalanceo de clases que afecta tanto a usuarios como a puntos de interés, así como la transferencia de conocimiento entre diferentes ciudades, mediante el aprendizaje por transferencia. El objetivo final es comprobar si los métodos y aplicaciones de transferencia propuestos se comportan adecuadamente ante diferentes tipos de datos.

III. Relevancia

La llegada del aprendizaje por transferencia abre todo un abanico de posibilidades en el campo del aprendizaje automático. Aporta el potencial de resolver problemas que hasta ahora estaban fuera de nuestro alcance, ya sea por la falta de datos etiquetados o por los enormes recursos necesarios para recopilarlos. A pesar de partir de sistemas de recomendación como principal caso de uso, la intención es desarrollar métodos generales de aprendizaje por transferencia que puedan aplicarse con éxito a otras tareas de aprendizaje automático. Por ejemplo, el objetivo del *feature-based transfer learning* es buscar nuevas representaciones de características haciendo uso del conocimiento del dominio fuente. Se centra, por tanto, en la propia representación de los datos, siendo independiente de la tarea a realizar. Desde el punto de vista científico, una de las principales ventajas de elegir sistemas de recomendación basados en contenido como ejemplo de aplicación reside en los tipos de datos con los que trabaja, es decir, imágenes y comentarios de texto. Por tanto, el estudio del aprendizaje por transferencia en este contexto podría tener un gran impacto en dos áreas muy importantes de aplicación del aprendizaje automático, visión artificial y procesamiento de lenguaje natural.

Dado que una parte importante de nuestra investigación se centra en el análisis y la mejora de sistemas de recomendación, los resultados podrían tener un impacto directo en la economía. En lo que respecta a los servicios de alojamiento y alimentación, se estima que la inteligencia artificial tiene el potencial de aumentar las tasas de rentabilidad en alrededor de un 74% para el año 2035 [8]. Por tanto, invertir en este aspecto podría ser muy productivo. En cuanto al turismo, con servicios de recomendación personalizada, un visitante podría tener acceso a un mayor número de propuestas y recibir la más adecuada según sus preferencias. De esta forma se podría extraer el máximo potencial de los recursos de un destino, ya que se trata de ofrecer la mejor opción entre todas las posibilidades. En cierto sentido, cuando se utiliza un sistema de recomendación para promocionar la oferta turística o gastronómica de una región de forma personalizada, el objetivo es potenciar los puntos fuertes, recomendando lo que creemos que mejor se adapta a las preferencias del usuario concreto, e intentar que los puntos débiles pasen desapercibidos, ignorando aquellas opciones que creemos que tendrían un efecto negativo. En consecuencia, la investigación en este campo permite desarrollar herramientas dedicadas a conseguir la forma óptima de sacar el máximo partido a la oferta turística y gastronómica de una región, extrayendo su máximo potencial. Además, es importante destacar que una recomendación personalizada se basa en los gustos del usuario, no en la popularidad del objeto a recomendar. Así, se posibilita el descubrimiento de nuevos productos, servicios, experiencias o lugares, promoviendo un turismo de mayor calidad y con un abanico más amplio de posibilidades. Esto también aumenta considerablemente las oportunidades de crecimiento de nuevas propuestas emergentes.

Por último, una de nuestras líneas de trabajo consiste en detectar factores relevantes en los comentarios de los usuarios, lo que tiene ventajas muy importantes en varios sentidos. Para empezar, la identificación de los puntos fuertes y débiles de un punto de interés (o de un restaurante) permite detectar tanto las deficiencias mejorables como los aspectos favorables que se pueden potenciar. Esto no sólo conduce a una mejora de la calidad turística, sino también a un aumento del número de visitantes, ya que esta mejora se reflejará en la percepción final del turista y, en consecuencia, en la imagen que proyectará del lugar. Además, detectar cuáles de estos factores relevantes lo son también para un usuario concreto permite ofrecer una recomendación personalizada más eficaz. La importancia de una buena recomendación es evidente en la percepción final del visitante, por lo que, de nuevo, podría tener un importante impacto en el sector turístico.

REFERENCES

[1] Eva Blanco-Mallo, Laura Morán-Fernández, Beatriz Remeseiro, and Verónica Bolon Canedo. Do all roads lead to Rome? Studying distance measures in the context of machine learning. *Submitted to Artificial Intelligence Review*, 2021.

[2] Eva Blanco-Mallo, Beatriz Remeseiro, Verónica Bolón-Canedo, and Amparo Alonso-Betanzos. On the effectiveness of convolutional autoencoders on image-based personalized recommender systems. In *Multidisciplinary Digital Publishing Institute Proceedings*, volume 54, page 11, 2020.

[3] Eva Blanco-Mallo, Beatriz Remeseiro, Verónica Bolon Canedo, Amparo Alonso-Betanzos, et al. Only for your eyes: Visualizing the results of a personalized recommendation system. In *Workshop ECAI 2020 - Art and artificial intelligence: a disruptive joint venture*, 2020.

[4] Francois Chollet. Building Autoencoders in Keras. https://blog.keras.io/building-autoencoders-in-keras.html, 2016.

[5] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[6] Jorge Díez, Pablo Pérez-Núñez, Oscar Luaces, Beatriz Remeseiro, and Antonio Bahamonde. Towards explainable personalized recommendations by learning from users' photos. *Information Sciences*, 520:416–430, 2020.

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[8] Mark Purdy and Paul Daugherty. How AI boosts industry profits and innovation, 2017.

Algoritmos heurísticos para problemas de embebido de grafos en disposiciones circulares

Sergio Cavero
Universidad Rey Juan Carlos
Móstoles, Madrid, España
sergio.cavero@urjc.es

Datos de la Tesis Doctoral

- Directores: Abraham Duarte (abraham.duarte@urjc.es) y Eduardo G. Pardo (eduardo.pardo@urjc.es).
- Departamento: Ciencias de la Computación, Arquitectura de Computadores, Lenguajes y Sistemas Informáticos y Estadística e Investigación Operativa.
- Institución: Universidad Rey Juan Carlos.
- Fecha de inicio: septiembre de 2020.

***Resumen* — La optimización es una disciplina relacionada con las matemáticas y la ciencia de la computación que aborda la búsqueda de la mejor solución posible, denominada solución óptima, a un problema modelado matemáticamente. Para algunos de estos problemas, especialmente los pertenecientes a la clase NP-Difícil, obtener la solución óptima supone una gran dificultad computacional y, en ocasiones, el tiempo necesario para encontrarla es inasumible. Para poder abordar este tipo de problemas se utilizan técnicas aproximadas que permiten encontrar soluciones de calidad, aunque no necesariamente óptimas, como por ejemplo los algoritmos heurísticos y metaheurísticos. Los algoritmos heurísticos se caracterizan por ser procedimientos simples, basados en ideas sencillas. Los algoritmos metaheurísticos, son algoritmos más complejos que guían a los algoritmos heurísticos en la búsqueda de la solución óptima. La familia de problemas de optimización abordada en esta Tesis Doctoral, utilizando algoritmos heurísticos y metaheurísticos, engloba diferentes problemas que son modelados matemáticamente mediante el uso de grafos. Más concretamente, los conjuntos de problemas de optimización estudiados presentan la característica común de proyectar grafos con cualquier estructura, en grafos con estructura de tipo ciclo, de modo que se optimice una determinada función objetivo.**

***Palabras clave* —** embebido de grafos, grafo ciclo, heurística, metaheurística, optimización

I. Introducción

El desarrollo de esta Tesis Doctoral se enmarca en la disciplina de la optimización heurística. Esta disciplina aborda la búsqueda de la mejor solución posible a un problema de optimización de manera aproximada. Un problema de optimización se define como la maximización o minimización de una o varias funciones objetivo donde las soluciones satisfacen un conjunto de restricciones. Además, la optimización heurística tiene gran interés en la resolución de problemas computacionalmente difíciles (como son aquellos pertenecientes a la clase de complejidad NP-Difícil) con aplicación práctica en industria, transporte, ingeniería y otras disciplinas. El tratamiento de estos problemas requiere un modelado previo para poder ser tratados computacionalmente. En este sentido, muchos de los problemas de optimización son modelados mediante grafos, estructuras de datos comúnmente utilizadas en medios computacionales para modelar sistemas de la vida real y otras abstracciones.

Entre los problemas de optimización modelados mediante grafos se encuentra una subfamilia de problemas, denominados problemas de embebido o etiquetado de grafos (GLP, del inglés, *Graph Layout Problems*) [13], los cuales han sido ampliamente estudiados en la literatura debido al gran número de aplicaciones prácticas que tienen. Ejemplo de estas aplicaciones son: el diseño de circuitos, gestión del espectro de retransmisión de redes inalámbricas, dibujado de grafos o la migración de redes de telecomunicaciones [26]. En la Sección III se discute la relevancia de estos problemas, así como su posible aplicación práctica y transferencia a la sociedad.

El objetivo de los GLP es optimizar una determinada función objetivo cuando se realiza una proyección de un grafo candidato, $G_C = (V_C, E_C)$, en un grafo huésped, $G_H = (V_H, E_H)$, donde V_C y V_H representan el conjunto de vértices y E_C y E_H representan al conjunto de aristas. Esta proyección es generalmente conocida como embebido o etiquetado y consiste en definir una función biyectiva que relaciona los vértices del conjunto V_G con los vértices del conjunto V_H. Formalmente, se define una función φ tal que $\varphi : V_C \rightarrow V_H$. Habitualmente, los GLP con más interés son aquellos en los que el grafo G_H tiene una estructura regular. En concreto, las estructuras regulares más utilizadas en la literatura son: los caminos, los ciclos, los árboles y las rejillas [13].

Esta Tesis Doctoral se centra en el estudio de problemas pertenecientes a la familia de los GLP, donde el grafo huésped es un ciclo. Estos grafos satisfacen ser eulerianos, hamiltonianos, de distancia unitaria y regulares de grado 2, tal que $|V_H| = |E_H| = |V_G|$. En la Figura 1a se muestra un ejemplo de un grafo candidato $G_C = (V_C, E_C)$, con $V_C = \{\mathrm{A}, \mathrm{B}, \mathrm{C}, \mathrm{D}, \mathrm{E}, \mathrm{F}\}$ y $E_C = \{(\mathrm{A}, \mathrm{B}), (\mathrm{B}, \mathrm{C}), (\mathrm{B}, \mathrm{D}), (\mathrm{C}, \mathrm{F}), (\mathrm{D}, \mathrm{E}), (\mathrm{D}, \mathrm{F})\}$. En la Figura 1b se muestra un grafo ciclo que actúa como huésped y, finalmente, en la Figura 1c se ilustra un posible embebido de G_C

en G_H. Más formalmente, la función φ es definida en este caso por la siguiente proyección: $\varphi(\mathrm{A}) = 1$, $\varphi(\mathrm{B}) = 2$, $\varphi(\mathrm{C}) = 3$, $\varphi(\mathrm{D}) = 6$, $\varphi(\mathrm{E}) = 5$ y $\varphi(\mathrm{F}) = 4$.

En esta Tesis Doctoral se ha optado por el estudio de una subfamilia de los GLP donde el grafo huésped tiene una estructura de ciclo. De esta manera, al centrar la Tesis en un conjunto problemas que guardan relación es posible detectar propiedades similares y aplicar el conocimiento adquirido a medida que avanza el proceso de investigación. Además, las estrategias o algoritmos propuestos pueden ser adaptados con mayor facilidad a problemas pertenecientes a la misma familia.

Para determinar los problemas a abordar en la Tesis Doctoral se ha realizado un estudio previo del estado del arte de todos los problemas enmarcados en la familia de los GLP cuyo grafo huésped es un ciclo. Posteriormente, se han seleccionado aquellos problemas con mayor interés y relevancia en la literatura. Considerando el tiempo necesario para la realización de la Tesis Doctoral se propone el estudio de cuatro problemas de optimización.

La principal diferencia entre los cuatro problemas seleccionados es su función objetivo. Esta función objetivo suele requerir del cálculo de la distancia a la que se encuentran dos vértices adyacentes del grafo de entrada, medida en el ciclo. Para ello, dado una arista, $(u, v) \in E_G$, se calcula la cardinalidad de los caminos con inicio en $\varphi(u)$ y fin en $\varphi(v)$, o bien, con inicio en $\varphi(v)$ y fin en $\varphi(u)$. La menor cardinalidad de todos los posibles caminos se considera la distancia entre los vértices u y v. En este trabajo, se denota formalmente a la distancia de dos vértices $u, v \in V_G$, tal que $(u, v) \in E_G$, como $d(u, v)$. Por ejemplo, partiendo de la Figura 1c, la distancia entre los vértices adyacentes B y D es 2, puesto la menor cardinalidad entre todos los caminos, con inicio en $\varphi(\mathrm{B})$ y fin en $\varphi(\mathrm{D})$, o viceversa, es 2:

$$\begin{aligned} d(\mathrm{B},\mathrm{D}) = &\min\{|\{(2,3),(3,4),(4,5),(5,6)\}|, \\ &|\{(6,1),(1,2)\}|\} = \min\{4,2\} = 2. \end{aligned} \tag{1}$$

Considerando las anteriores definiciones, los problemas de optimización planteados a estudio en esta Tesis Doctoral son los siguientes:

- *Cyclic Antibandwidth Problem* (CAP): consiste en la maximización de la distancia a la que se encuentran los vértices adyacentes, medida en el ciclo [2, 5, 14, 23, 24, 25, 28, 27, 40, 41]. Más formalmente, el problema se puede formular como:

$$\mathrm{CAP} = \arg\max_{\varphi\in\Phi}\{\arg\min_{(u,v)\in E_G}\{d(u,v)\}\} \tag{2}$$

- *Cyclic Bandwidth Problem* (CBP): consiste en la minimización de la distancia a la que se encuentran los vértices adyacentes, medida en el ciclo [10, 18, 29, 34, 44]. Más formalmente, el problema se puede formular como:

$$\mathrm{CBP} = \arg\min_{\varphi\in\Phi}\{\arg\max_{(u,v)\in E_G}\{d(u,v)\}\} \tag{3}$$

- *Cyclic Bandwidth Sum Problem* (CBSP): consiste en la minimización de la suma de la distancia a la que se encuentran los vértices adyacentes, medida en el ciclo [11, 21, 17, 33, 32, 31, 35, 43]. Más formalmente, el problema se puede formular como:

$$\mathrm{CBSP} = \arg\min_{\varphi\in\Phi}\{\sum_{(u,v)\in E_G} d(u,v)\} \tag{4}$$

- *Cyclic Cutwidth Minimization Problem* (CCMP): consiste en la minimización del número de aristas del grafo candidato que concurren con una arista del grafo huésped [1, 4, 6, 9, 15, 19, 20, 22, 30, 36, 37, 38]. Más formalmente, el problema se puede formular como:

$$\mathrm{CCMP} = \arg\min_{\varphi\in\Phi}\{\arg\max_{(w,z)\in E_H}\{\mathit{cut}(w,z,\varphi)\}\}, \tag{5}$$

  donde $\mathit{cut}(w, z, \varphi)$ es una función que calcula el número de aristas del grafo de entrada que concurren en la arista (w, z) del grafo huésped.

De manera más concreta, una vez identificado el conjunto de problemas que se pretenden abordar en esta Tesis Doctoral, se enuncia una hipótesis general válida para todos ellos. Dicha hipótesis es una propuesta tentativa que pretende formular una solución a los problemas abordados. Esta hipótesis representará un elemento fundamental en el proceso de investigación, ya que servirá de guía del mismo. En este caso, la hipótesis que se plantea para el desarrollo de esta Tesis se puede resumir en los siguientes términos:

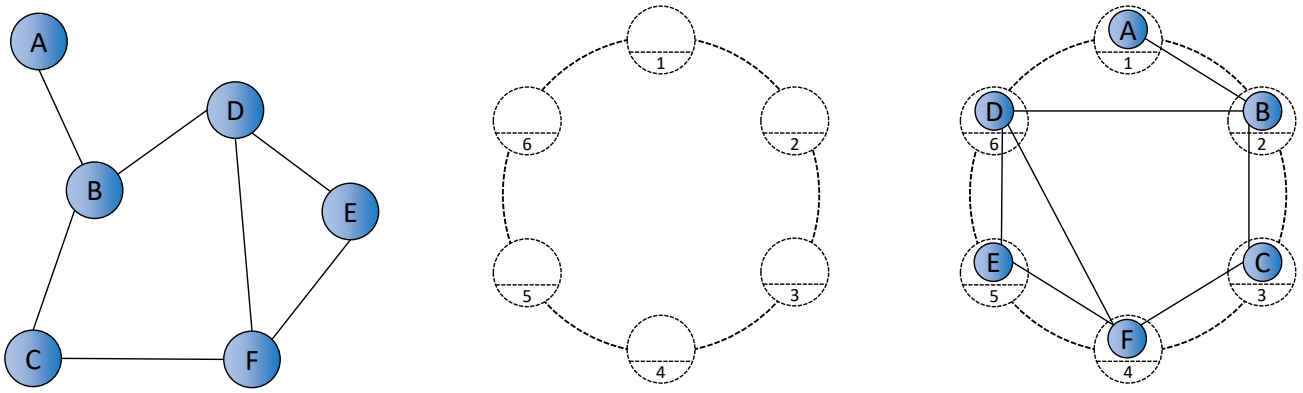
> "*Los algoritmos heurísticos, utilizados junto con técnicas metaheurísticas, son métodos capaces de encontrar soluciones de alta calidad, potencialmente próximas a las óptimas, en problemas de optimización modelados mediante grafos consistentes en el embebido de un grafo candidato en un grafo huésped con estructura circular*".

Derivado de la hipótesis anterior, el objetivo principal de esta Tesis Doctoral se enuncia de la siguiente manera:

> "*Diseñar e implementar algoritmos heurísticos, utilizados junto con técnicas metaheurísticas, para abordar problemas de optimización modelados mediante grafos consistentes en el embebido de grafos en ciclos*".

Para alcanzar el objetivo principal planteado, es necesario cubrir los siguientes objetivos específicos:

1) Realizar un estudio del estado del arte de cada problema.
2) Estudiar de manera exhaustiva las características estructurales de cada problema.
3) Modelar cada uno de los problemas para poder abordarlo computacionalmente.
4) Identificar estrategias y propiedades comunes que puedan ser de aplicabilidad general para problemas similares.
5) Proponer, diseñar e implementar estrategias heurísticas para cada problema.
6) Escoger, parametrizar e implementar el algoritmo metaheurístico más adecuado, que permita mejorar las soluciones encontradas por métodos previos.



(a) (b) (c)

Fig. 1: (a) Un grafo de candidato, G_C. (b) Un grafo huésped, G_H. (c) Un ejemplo de una proyección o embebido del grafo G_C en G_H.

7) Comparar los resultados obtenidos por los algoritmos propuestos, con los algoritmos previos del estado del arte, sobre conjuntos de datos comunes.
8) Elaborar una memoria que recoja el trabajo realizado, conclusiones y posibles mejoras.
9) Diseminar los resultados encontrados mediante publicaciones científicas relevantes.

A continuación, se detalla la metodología y el plan de trabajo escogido que permitan la consecución de los objetivos establecidos.

II. Metodología y plan de trabajo

El desarrollo de esta Tesis Doctoral se fundamenta en el método científico como metodología para generar nuevo conocimiento. Concretamente, este método se basa en la observación, medición, experimentación, formulación y modificación de una hipótesis. Además, este método puede ser aplicado a cualquier área científica cuyo fin sea el de aportar conocimiento nuevo. En este caso, el método científico es utilizado como guía para proponer algoritmos heurísticos y metaheurísticos para abordar un problema de optimización combinatoria como son los pertenecientes a la familia de los GLP. En la Figura 2, se representa el proceso de investigación propuesto en la Tesis Doctoral.

El proceso comienza con el planteamiento del problema (fase 1). A continuación, se lleva a cabo el estudio del estado del arte, es decir, conocer los avances más recientes del problema en cuestión (fase 2). En esta misma etapa se destacan las tareas de identificación y reproducción de los algoritmos previos y obtención de las instancias (también conocidas como entradas o casos de prueba) puesto que son fundamentales para el correcto desarrollo de la investigación.

Las siguientes cinco fases del proceso de investigación, detallado en la Figura 2, son repetidas iterativamente (fases 3, 4, 5, 6 y 7). Partiendo del planteamiento de una hipótesis (fase 3), se realiza un estudio del problema que permita la extracción de características y su modelado (fase 4). A continuación, se propone y se implementa un algoritmo para abordar el problema en cuestión. Generalmente, los algoritmos implementados serán: algoritmos heurísticos, algoritmos metaheurísticos, o estrategias avanzadas que aumenten la eficiencia o robustez del algoritmo propuesto (fase 5). Tras la implementación de un algoritmo es necesario analizar su rendimiento mediante un conjunto de experimentos (fase 6). Estos experimentos son clasificados en: experimentos preliminares, si el objetivo es establecer la mejor configuración de los parámetros del algoritmo propuesto; o bien en experimentos competitivos, destinados a comparar el algoritmo propuesto con los algoritmos del estado del arte. Finalmente, se valida la hipótesis de partida (fase 7). Los resultados más relevantes de la investigación realizada son diseminados por medio de publicaciones en revistas o conferencias o *workshops* (fase 8).

Por último, se recalca que cada una de las fases del proceso de investigación descrito en la Figura 2 se relacionan con uno o varios de los objetivos planteados en la Sección I. Concretamente las fases 1 y 2 se corresponden con el objetivo 1. De manera similar, el objetivo 2 se corresponde con la fase 4, los objetivos 5, 6 y 7 con la fase 5, el objetivo 7 con la fase 6 y, finalmente, los objetivos 8 y 9 se corresponden con la fase 8. Las fases 3 y 7, consistentes en el planteamiento y validación de la hipótesis, condicionan el diseño general del proceso de investigación.

III. Relevancia

El trabajo de investigación que se pretende desarrollar en la Tesis Doctoral se enmarca en la ciencia básica, es decir, la investigación realizada no se lleva a cabo por un fin práctico inmediato, sino con el fin de aportar conocimiento nuevo. No obstante, los problemas abordados en esta investigación han sido relacionados con múltiples aplicaciones prácticas. Algunas de ellas se recogen a continuación: el diseño de circuitos integrados a gran escala (VLSI, del inglés, *Very Large-Scale Integration* [12]; la gestión del espectro de retransmisión de redes inalámbricas [16]; la localización de instalaciones en una

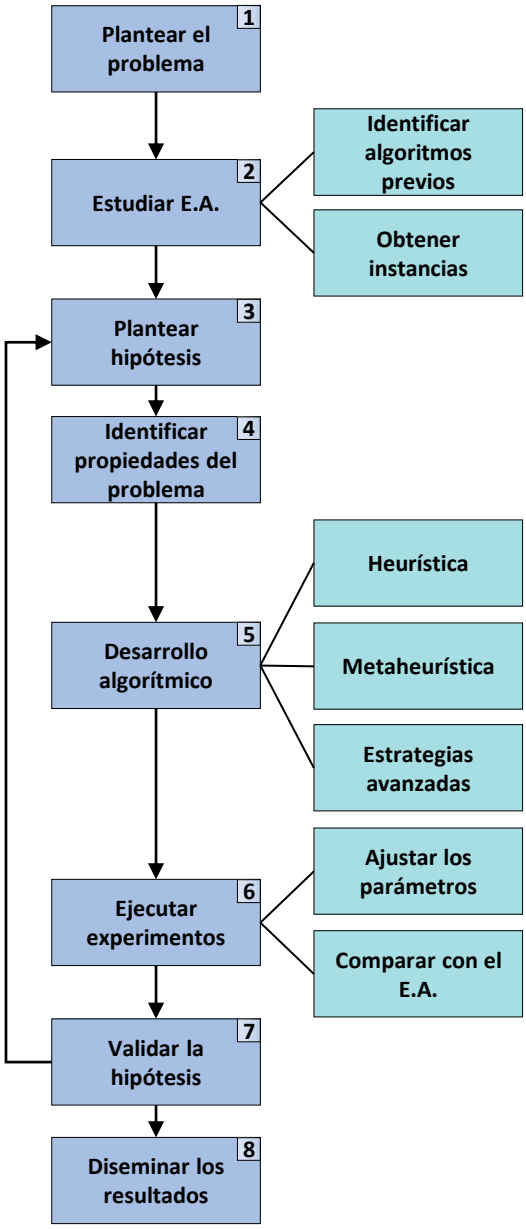


Fig. 2: Representación de la adaptación del método científico al contexto de esta Tesis Doctoral.

población [42]; el dibujado de grafos (*Graph Drawing Problems*) [39]; la planificación de tareas en un multiprocesador (*Multiprocessor Scheduling Problems*) [23]; o la migración de una red de telecomunicaciones (*Network Migration Scheduling Problems*) [3].

Desde el punto de vista académico, la investigación es un proceso en el que colabora toda la comunidad científica, compartiendo sus conocimientos para que puedan servir de ayuda en las investigaciones de otros. Por lo tanto, se espera que los algoritmos propuestos, así como cualquier otra contribución, pueda ser de utilidad para la comunidad científica. Además, todos los resultados obtenidos han sido publicados en los correspondientes artículos y se espera que faciliten futuras comparaciones.

Por último, los resultados parciales obtenidos hasta el momento remarcan la relevancia de la investigación realizada en esta Tesis Doctoral. Concretamente, se ha publicado un artículo [9] en la revista *Computers & Operations Research*, indexada en el índice *Journal of Citation Reports* (JCR) y situada en el segundo cuartil (Q2). Concretamente, en este artículo se aborda uno de los cuatro problemas descritos en la Sección I. Por otro lado, se encuentra en estado de revisión un artículo en el que se trabaja otro de los problemas descritos en la Sección I. En este caso, el artículo ha sido enviado a la revista *Computational Optimization and Applications*, indexada en el primer cuartil en el índice JCR. Cabe destacar que el desarrollo de los dos trabajos anteriores ha producido resultados preliminares que han sido publicados en un congreso internacional [7], en una revista indexada en el *SCImago Journal & Country* [8] y en diferentes talleres nacionales, entre los que se destaca la II Escuela de Invierno de la Red Española de Optimización Heurística (Red HEUR).

Agradecimientos

Esta investigación ha sido parcialmente apoyada por el Ministerio de Ciencia, Innovación y Universidades (Ref. PGC2018-095322-B-C22 y Ref. FPU19/04098) y por la Comunidad de Madrid y el Fondo Europeo de Desarrollo Regional (Ref. P2018/TCS-4566).

Bibliografía

[1] H.L. Abbott. “Hamiltonian circuits and paths on the n-cube”. In: *Canadian Mathematical Bulletin* 9.5 (1966), pp. 557–562.

[2] G.G. Alway and D.W. Martin. “An algorithm for reducing the bandwidth of a matrix of symmetrical configuration”. In: *The Computer Journal* 8.3 (1965), pp. 264–272.

[3] D.V. Andrade and M.G.C. Resende. “GRASP with path-relinking for network migration scheduling”. In: *Proceedings of the International Network Optimization Conference*. 2007, pp. 1–7.

[4] R. Aschenbrenner. “A Proof for the Cyclic Cutwidth of Q5”. In: *REU Project, Cal State Univ., San Bernardino* (2001).

[5] R. Bansal and K. Srivastava. “A memetic algorithm for the cyclic antibandwidth maximization problem”. en. In: *Soft Computing* 15.2 (Feb. 2011), pp. 397–412. ISSN: 1433-7479.

[6] C. Castillo. “A Proof for the Cyclic Cutwidth of Q6”. In: *REU Project, Cal State Univ., San Bernardino* (2003).

[7] S. Cavero and E.G. Pardo. “A VNS approach for a variant of the Antibandwidth problem”. In: *International Conference on Variable Neighborhood Search (ICVNS)*. 2021.

[8] S. Cavero, E.G. Pardo, and A. Duarte. "A VNS approach for a variant of the Antibandwidth problem". In: *The Conference of the Spanish Association of Artificial Intelligence (CAEPIA)*. Lecture Notes in Artificial Intelligence. 2021.

[9] S. Cavero, E.G. Pardo, M. Laguna, and A. Duarte. "Multistart search for the Cyclic Cutwidth Minimization Problem". In: *Computers & Operations Research* 126 (2021), pp. 105–116. ISSN: 0305-0548.

[10] W.H. Chan, P.C.B. Lam, and W.C. Shiu. "Cyclic bandwidth with an edge added". In: *Discrete applied mathematics* 156.1 (2008), pp. 131–137.

[11] Y. Chen and J. Yan. "A study on cyclic bandwidth sum". en. In: *Journal of Combinatorial Optimization* 14.2 (Oct. 2007), pp. 295–308. ISSN: 1573-2886.

[12] J. Cohoon and S. Sahni. "Heuristics for backplane ordering". In: (1985).

[13] J. Díaz, J. Petit, and M. Serna. "A survey of graph layout problems". In: *ACM Computing Surveys (CSUR)* 34.3 (2002), pp. 313–356.

[14] Stefan Dobrev, Rastislav Královič, Dana Pardubská, L'ubomír Török, and Imrich Vrt'o. "Antibandwidth and cyclic antibandwidth of Hamming graphs". In: *Discrete Applied Mathematics* 161.10 (July 2013), pp. 1402–1408. ISSN: 0166-218X.

[15] Jason Erbele, Joseph Chavez, and Rolland Trapp. "The cyclic cutwidth of Qn". In: *Manuscript, California State University, San Bernardino USA* (2003).

[16] W.K. Hale. "Frequency assignment: Theory and applications". In: *Proceedings of the IEEE* 68.12 (1980), pp. 1497–1514.

[17] R. Hamon, P. Borgnat, P. Flandrin, and C. Robardet. "Relabelling vertices according to the network structure by minimizing the cyclic bandwidth sum". In: *Journal of Complex Networks* 4.4 (Dec. 2016). Publisher: Oxford Academic, pp. 534–560. ISSN: 2051-1310.

[18] L.H. Harper. "Optimal assignments of numbers to vertices". In: *Journal of the Society for Industrial and Applied Mathematics* 12.1 (1964), pp. 131–135.

[19] P. Jain, K. Srivastava, and G. Saran. "Minimizing cyclic cutwidth of graphs using a memetic algorithm". In: *Journal of Heuristics* 22.6 (2016), pp. 815–848.

[20] B. James. "The Cyclical Cutwidth of the Three-dimensional and Fourdimensional Cubes". In: *Cal State Univ., San Bernardino McNair Scholar's Program Summer Research Journal* (1996).

[21] H. Jianxiu. "Cyclic bandwidth sum of graphs". en. In: *Applied Mathematics-A Journal of Chinese Universities* 16.2 (June 2001), pp. 115–121. ISSN: 1993-0445.

[22] M. Johnson. "The linear and cyclic cutwidth of the complete bipartite graph". In: *REU Project, Cal State Univ., San Bernardino* (2003).

[23] J.Y.T. Leung, O. Vornberger, and J.D. Witthoff. "On some variants of the bandwidth minimization problem". In: *SIAM Journal on Computing* 13.3 (1984), pp. 650–667.

[24] M. Lozano, A. Duarte, F. Gortázar, and R. Martí. "A hybrid metaheuristic for the cyclic antibandwidth problem". In: *Knowledge-Based Systems* 54 (2013), pp. 103–113. ISSN: 0950-7051.

[25] Z. Miller and D. Pritikin. "On the separation number of a graph". In: *Networks* 19.6 (1989), pp. 651–666. ISSN: 1097-0037.

[26] E.G. Pardo, R. Martí, and A. Duarte. "Linear Layout Problems". In: *Handbook of Heuristics*. Ed. by R. Martí, P. Panos, and M.G.C. Resende. Springer International Publishing, 2016, pp. 1–25. ISBN: 978-3-319-07153-4.

[27] A. Raspaud, H. Schröder, O. Sykora, L. Torok, and I. Vrt'o. "Antibandwidth and cyclic antibandwidth of meshes and hypercubes". en. In: *Discrete Mathematics*. 7th International Colloquium on Graph Theory 309.11 (June 2009), pp. 3541–3552. ISSN: 0012-365X.

[28] A. Raspaud, O. Sykora, and I. Vrt'o. "Congestion and Dilation, Similarities and Differences: a Survey". en. In: *Proceedings of the 7th International Colloquium on Structural Information and Communication Complexity*. 2000, p. 14.

[29] J. Ren, J. Hao, and E. Rodriguez-Tello. "A study of recombination operators for the cyclic bandwidth problem". In: *International Conference on Artificial Evolution (Evolution Artificielle)*. Springer. 2019, pp. 177–191.

[30] F.R. Rios. "Complete graphs as a first step toward finding the cyclic cutwidth of the n-cube". In: *Cal State Univ., San Bernardino McNair Scholar's Program Summer Research Journal* (1996).

[31] E. Rodriguez-Tello, F. Lardeux, A. Duarte, and V. Narvaez-Teran. "Alternative evaluation functions for the cyclic bandwidth sum problem". en. In: *European Journal of Operational Research* 273.3 (Mar. 2019), pp. 904–919. ISSN: 0377-2217.

[32] E. Rodriguez-Tello, V. Narvaez-Teran, and F. Lardeux. "Comparative Study of Different Memetic Algorithm Configurations for the Cyclic Bandwidth Sum Problem". In: *Parallel Problem Solving from Nature – PPSN XV*. Ed. by Anne Auger, Carlos M. Fonseca, Nuno Lourenço, Penousal Machado, Luís Paquete, and Darrell Whitley. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2018, pp. 82–94. ISBN: 978-3-319-99253-2.

[33] E. Rodriguez-Tello, V. Narvaez-Teran, and F. Lardeux. "Dynamic Multi-Armed Bandit Algorithm for the Cyclic Bandwidth Sum Problem". In: *IEEE Access* 7 (2019), pp. 40258–40270. ISSN: 2169-3536.

[34] E. Rodriguez-Tello, H. Romero-Monsivais, G. Ramirez-Torres, and F. Lardeux. "Tabu search for the cyclic bandwidth problem". In: *Computers & Operations Research* 57 (2015), pp. 17–32.

[35] D. Satsangi, K. Srivastava, and G. "General variable neighbourhood search for cyclic bandwidth sum minimization problem". In: *2012 Students Conference on Engineering and Systems*. 2012, pp. 1–6.

[36] H. Schröder, O. Sykora, and I. Vrt'o. "Cyclic cutwidth of the mesh". In: *International Conference on Current Trends in Theory and Practice of Computer Science*. Springer. 1999, pp. 449–458.

[37] H. Schröder, O. Sykora, and I. Vrt'o. "Cyclic cutwidths of the two-dimensional ordinary and cylindrical meshes". In: *Discrete applied mathematics* 143.1-3 (2004), pp. 123–129.

[38] V. Sciortino, J. Chavez, and R. Trapp. "The Cyclic Cutwidth of a $P_2 \times P_2 \times P_n$ Mesh". In: *REU Project, Cal State Univ., San Bernardino* (2002).

[39] F. Shahrokhi, O. Sykora, L.A. Székely, and I. Vrto. "On bipartite drawings and the linear arrangement problem". In: *SIAM Journal on Computing* 30.6 (2001), pp. 1773–1789.

[40] S.S. Skiena. "Graph Traversal". In: *The Algorithm Design Manual*. 2nd. Springer Publishing Company, Incorporated, 1997. ISBN: 978-1-84800-069-8.

[41] O. Sykora, L. Torok, and I. Vrt'o. "The Cyclic Antibandwidth Problem". In: *Electronic Notes in Discrete Mathematics*. 7th International Colloquium on Graph Theory 22 (2005), pp. 223–227. ISSN: 1571-0653.

[42] A. Tamir. "Obnoxious facility location on graphs". In: *SIAM Journal on Discrete Mathematics* 4.4 (1991), pp. 550–567.

[43] J. Yuan. *Cyclic arrangement of graphs*. Graph Theory Notes of New York. 1995.

[44] S. Zhou. "Bounding the bandwidths for graphs". In: *Theoretical computer science* 249.2 (2000), pp. 357–368.

Modelado y análisis de la marcha humana mediante sensores inerciales y aprendizaje profundo.

1st Diego Teran P.
University of Malaga
Málaga, España
Estudiante de Doctorado
dfteran@uma.es

2nd Enrique Dominguez
Dept. of Computer Sciencce
University of Malaga
Málaga, España
Director
enriqued@lcc.uma.es

***Abstract*—El análisis de la marcha es ampliamente utilizado por los médicos para detectar anomalías y concluir posibles tratamientos para el paciente. Convencionalmente, el análisis de la marcha se ha considerado de manera subjetiva por parte de los expertos sanitarios y ahora se utiliza la tecnología para mejorar la información de los datos. Sin embargo, la adquisición de esta información no es sencilla, ya que los ruidos de los sensores provocan errores en los datos cinemáticos para analizar cualquier forma de onda, y este análisis requiere una gran cantidad de datos (sin ruido) para el uso de inteligencia artificial. Con esta investigación se pretende desarrollar un sistema de adquisición portátil no invasivo capaz de procesar y filtrar las señales de los sensores en tiempo real, y además, obtener el modelado y análisis de la marcha humana utilizando técnicas de inteligencia artificial (aprendizaje profundo) y mediante la creación de un sistema para probablemente disminuir el número de sensores con el uso del aprendizaje del sistema.**

***Index Terms*—modelo de marcha humana, modelo cinemático, sensores inerciales, aprendizaje automático.**

I. Introduccion

A. Descripción de la investigación

La marcha humana es considerado una de las actividades más comunes y naturales, la cual involucra el esfuerzo combinado del cerebro, nervios y músculos [1]. Fisioterapeutas, ortopedistas y neurólogos examinan el movimiento humano para analizar y evaluar el estado, el tratamiento y la rehabilitación de un paciente [2].

El análisis de la marcha es el estudio de la locomoción humana que se realiza mediante: observación visual, tecnología de sensores, cámaras o integración de estas tecnologías [3]. La marcha humana es necesaria para comprender los movimientos de los humanos (cinemática) y las fuerzas (cinética) que se aplican en las articulaciones humanas [4].

La identificación del patrón de marcha normal, el diagnóstico médico, el cuidado geriátrico o de personas mayores, el control y las tácticas deportivas forman parte de la aplicación basada en el análisis de la marcha. La disponibilidad de parámetros cuantitativos de la marcha normal es esencial para la detección de trastornos de la marcha, así como: la identificación de las características del equilibrio, la evaluación de las intervenciones de la marcha médica y desarrollos de rehabilitación con la medición objetiva cuantitativa de los diferentes parámetros de la marcha y sobre la evolución y el diagnóstico precoz de diferentes enfermedades [2] [5]. Existen factores extrínsecos, intrínsecos, físicos psicológicos y patológicos que influyen la marcha normal, por lo que determinar los parámetros de marcha normal es una tarea muy compleja, siendo difícil el diagnóstico de la marcha [6].

En Estados Unidos, las caídas son la principal causa de muerte debido a lesiones no intencionales o no tratadas en adultos mayores (56.0% en 2014) [7]; además, se puede afirmar corroborativamente en otros países la causas de caidas por problemas de marcha en adultos mayores [2]: 33% de Australianos, 19.3% en Hong Kong, 21.6% en Bridgetown (Barbados), 29% en La Habana (Cuba), 33% en la Ciudad de México (México) y 34% en Santiago (Chile). Además, han logrado experimentar numerosos traumas o patologías si no se trata adecuadamente, para lo cual se hace necesario un sistema que permita al especialista conocer el modelo de la cinemática de la marcha humana del paciente para su rehabilitación y estudio.

El estudio de la marcha también es una herramienta útil para los profesionales de los deportes, debido a que pueden optimizar y aumentar el rendimiento deportivo. Además, los resultados pueden sugerir lesiones potenciales para tomar una acción preventiva, seleccionando las mejores opciones de tratamiento disponible y previniendo posibles lesiones deportivas [8] [9].

El estudio de la márcha se ha considerado converncionalmente a través de observaciones visuales subjetivas. En este trabajo se propone la creación de un sistema automático de modelado basado en aprendizaje profundo para estudiar la marcha, centrándonos en la articulación de la rodilla para conocer parámetros de la marcha [10] (velocidad de marcha, longitud de zancada y paso, fase de postura, fase de balanceo, marcha rápida de cadencia, simple y doble soporte, etc.), debido a que es la articulación más expuesta y menos protegida contra lesiones mecánicas como caidas.

B. Avances de la investigación

En este trabajo inicialmente pretendemos recrear la curva cinemática correspondiente a la rodilla en el plano sagital, para ello es necesario conocer el comportamiento de la rodilla en el

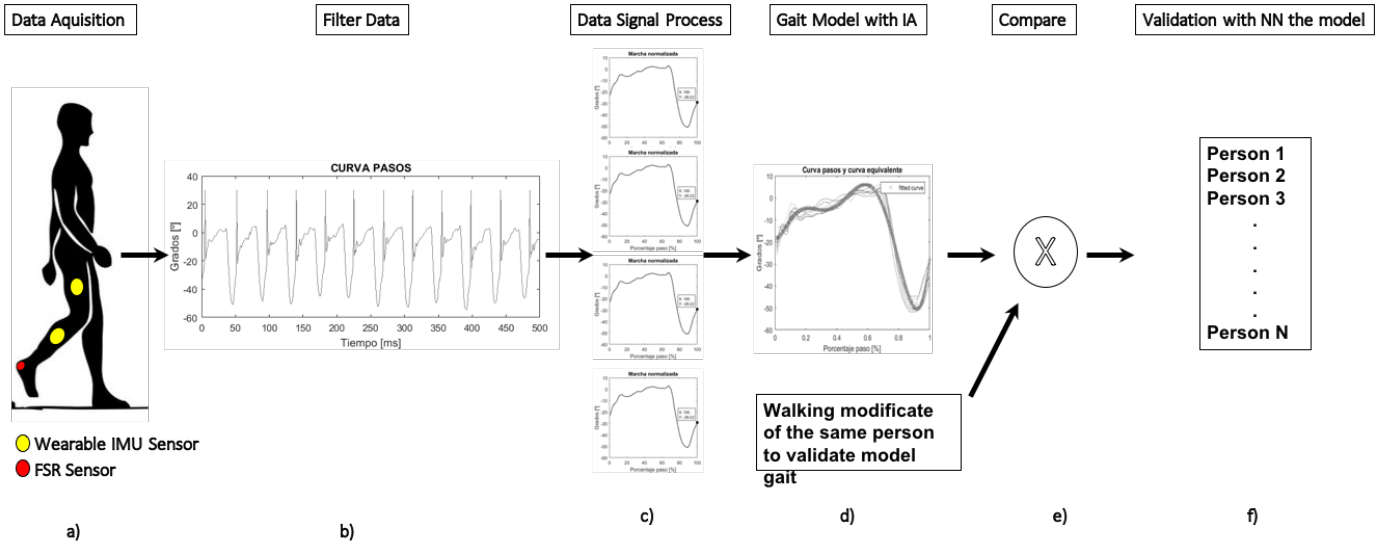


Fig. 1: Fuente de información para modelar la marcha humana [27]. a) Sistema de adquisición de datos . b) Pre-procesamiento (filtrado) de las señales. c) Procesamiento de datos (Clasificación de cada ciclo de marcha y normalización). d) Modelo de marcha. e) Modelos combinados. f) Validación del modelo con cada persona experimentada

plano mencionado y mostrar la curva cinemática. El ciclo de caminata comienza cuando el talón de un pie toca el suelo y termina con el contacto del suelo del mismo talón. En la figura 1 [27] se puede observar los pasos propuestos para abordar el problema.

Para la adquisición de datos de la marcha es necesario realizar un sistema de adquisición, para lo cual se desarrolló un sistema de adquisición portátil no invasivo utilizando un algoritmo recursivo en línea en un microcontrolador para procesar y filtrar señales de sensores portátiles. El sistema de señal de adquisición de datos (DAS) (Fig. 2) utiliza un resistor sensible a la fuerza (FSR) en el talón y dos sensores de inercia, uno en el muslo y otro en la pierna, para medir el ángulo de la rodilla; dicho sistema calibra automáticamente los sensores inerciales para cada experimento.

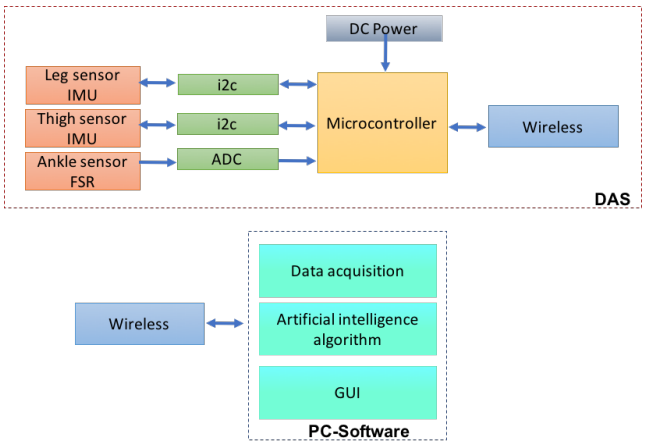


Fig. 2: Arquitectura del sistema.

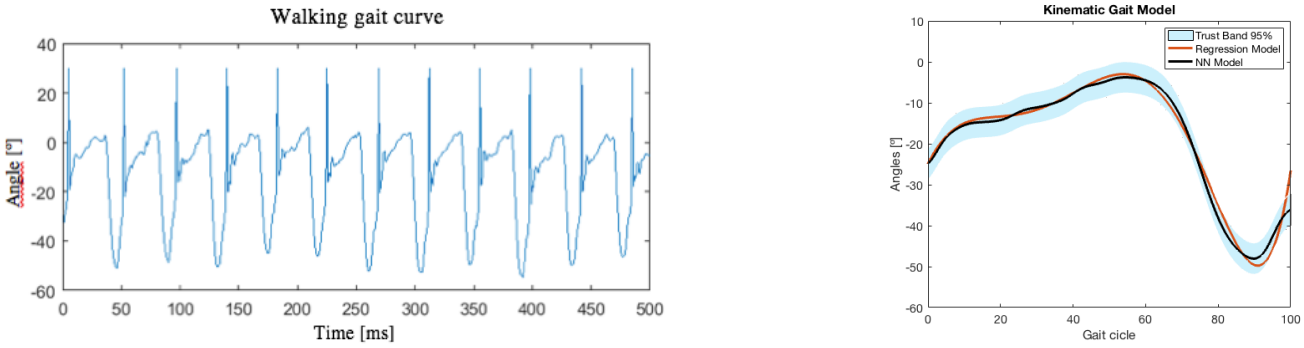
La figura 3a indica el resultado de adquirir los datos de la marcha por el sistema DAS; además, los pulsos que pasan de los 20º visibles en la Fig. 3a, es un dato enviado por el sistema para reconocer el inicio y el final de la marcha. En la Fig. 3 todos los pasos son función no lineal; por lo tanto, para estimar y obtener el modelo utilizamos el método de regresión y el modelo de red neuronal de aprendizaje supervisado para obtener los datos de ajuste. Los datos del vector se construyen usando los parámetros en la base de datos con un paso de velocidad normal, el vector tiene aproximadamente 750 datos (si los pacientes caminan lentamente, el vector tendrá más de 750 datos), y los datos establecidos se dividen aleatoriamente en tres grupos: entrenamiento, validación y prueba. Además, el peso y el sesgo se entrenarán con los algoritmos de gradiente conjugado de Levenberg, Bayesiano y Escalado para analizar el error. La figura 3b muestra el resultado de aplicar dichos modelos al primer paso (ciclo) de la marcha.

C. *Estado de arte*

En la literatura existe una gran cantidad de metodologías (basadas en procesamiento de imágenes, sensores de piso o colocados en cualquier parte del cuerpo, etc.) y discusiones sobre el reconocimiento y análisis de la marcha humana. Los dispositivos tecnológicos más utilizados en la investigación se pueden clasificar en [11]:

- Sensores no portables (non-wearable sensor - NWS)
- Sensores portables (wearable sensor - WS)
- Sistemas híbridos

Utilizar NWS requiere una sala con sensores instalados para capturar la marcha mientras el sujeto camina, lo cual implica un mayor coste de instalación y equipamiento. Para la captura de la marcha se han propuesto diferentes técnicas:



(a) (b)

Fig. 3: (a) Datos de la marcha adquiridos por wireless del sistema DAS (procesamiento de datos online), (b) Modelo cinemático de la marcha (Procesamiento de datos offline)

- Cámara triangular con 70% de precisión [12]
- Adquisición de video 3D mediante cámara en tiempo de vuelo (Time of flight - ToF) obteniendo un error de 2.66% - 9.25% [13].
- Luz de estructura, obteniendo menos de 1% (media) [14].
- Cámaras termo-gráficas con una exactitud de 78%-91% [15].

Los WS son sensores instalados en el cuerpo, no requieren un laboratorio y esta técnica adquiere la información de la marcha durante cada una de las actividades de la persona. Un sistema WS utiliza sensores como acelerómetros, sensores giroscópicos, magnetómetros, sensores de fuerza, extensómetros, goniómetros, marcadores activos o electromiografía [11] [5]. Utilizando sensores inerciales para obtener la orientación, la velocidad y la aceleración se ha determinado con gran precisión el ángulo de la articulación de flexión / extensión (coeficiente de correlación múltiple¿ 0,96) para todas las articulaciones [16] y con un error de zancada de -0.8 ±6.6 cm [17], Usando goniómetros se ha alcanzado una medida de error del 0.999 en estático [11]. En la actualidad, un 40% usan sistemas NWS y un 37.5% sistemas WS, aunque debido a su costo el porcentaje de sistemas WS usados en investigación aumenta [11] año tras año.

Los WS pueden almacenar datos durante la marcha o carrera [18], una limitación de tales métodos es que se necesita una cantidad significativa de experiencia y aportes humanos para procesar los datos, y, extraer características significativas para los análisis. Por lo tanto, puede ser ventajoso desarrollar enfoques avanzados de aprendizaje automático que no requieran una derivación a priori en las medidas de resultado [19]. Los sistemas WS tienen la ventaja de adquirir datos reales de una marcha sin modificación, lo que ocurre cuando se usa una cinta andadora que afecta la marcha o la carrera normal, afectando la precisión de análisis.

El inconveniente de los WS es la instalación y la incomodidad por el número de sensores utilizados para el estudio de la marcha. Igualmente, debido a la instalación con correas, puede provocar contener el funcionamiento de los músculos. Por otro lado, el coste computacional aumenta por el tratamiento de datos y del número de sensores usados. Además, los atletas prefieren correr sin nada unido a ellos, por lo tanto, existen investigaciones para estimar la marcha humana con el menor número de sensores posibles [20].

Los métodos de machine learning se han aplicado a una variedad de datos y tecnologías, la aplicación de las redes de aprendizaje profundo justifica la investigación en otros contextos, que pueden ser la biomecánica y el análisis del movimiento humano, en la cual se usan algoritmos como: Máquinas de soporte vectorial (SVM) para reconocimiento de actividades basadas en WS y árbol de decisión. Estos últimos son utilizados como clasificador para el reconocimiento de marcha [21] mediante datos adquiridos de sensores de inercia de smarth phone. Sin embargo, sigue siendo desconocido si las redes de aprendizaje profundo pueden contribuir a la tarea de reconocimiento de movimiento humano [19], sin embargo se tiene investigaciones actuales que usan WS para reconocimiento de marcha o actividades [22] [23].

Los patrones de marcha son aplicados en dos tareas: Identificación, para obtener la identidad del sujeto, y autenticación, para validar la identidad del sujeto. Existen técnicas que aplican redes de funciones de base radial (RBF) para aproximar las velocidades y aceleraciones e identificar sujetos [22]. En otra investigación realizan un sistema de reconocimiento de marcha de punta a punta basado en redes neuronales convolucionales(CNN) que usan datos de sensores inerciales a una velocidad a través del Bus PCI alrededor de $1\mu s$. Sin embargo, los modelos single task debe ejecutarse hasta tres veces para obtener los mismos resultados [22].

Se puede conocer investigaciones que usan técnicas de aprendizaje supervisado (Neural Nerwork, K-Nearest Neighbor, Supporting Vector Machine, Ensembles) y no supervisado (Self-Organizing Map, K means, Fuzzy c mean, hierarchinal), como redes neuronales profundas en espacio temporal [24] para dar a conocer métodos en aplicaciones de análisis de marcha [2], pero no se tiene información sobre aprendizaje automático de la marcha humana con el menor número de sensores en aplicaciones WS.

En este trabajo se plantea crear un sistema basado en

aprendizaje profundo que reconozca automáticamente el inicio y fin de la marcha humana mediante dos sensores inerciales. Dicho sistema se analizará en un entorno real con la ayuda de un especialista para analizar el comportamiento, el proyecto tiene como fin ayudar al médico especialista en tratamientos de traumas provocadas por lesiones o patologías que es muy común en la práctica clínica como lo da a conocer en [25] con la ayuda de algoritmos.

D. Objetivos

Cada vez gana más interés el estudio de la marcha humana usando técnica WS, no solo en el campo de la medicina, también en otros campos de investigación, como obtener el modelo de la cinemática para ayudar el estudio físico del sujeto. Es posible establecer diferentes técnicas de modelación de sistemas, así como la adquisición de datos mediante técnicas WS. Cabe recalcar que las pruebas en cintas de correr con NWS no dan datos reales debido a que afecta la caminata normal y en deportistas la carrera normal, por tanto, afecta a la precisión del análisis que en deportistas desean conocer [26] [20].

El objetivo principal que se pretende conseguir con este sistema es diseñar un sistema para modelar la cinemática de la marcha humana mediante la adquisición de datos de dos sensores inerciales. Para lo cual se pretende enseñar al sistema por medio de aprendizaje automático la detección de inicio y fin de la marcha; además, mediante aprendizaje profundo encontrar un modelo de marcha del sujeto.

Para la realización de este trabajo es necesario llevar a cabo otros objetivos más específicos que lleven a la consecuencia del objetivo final, entre ellos:

- Conocer nuevas técnicas de las tendencias en el desarrollo de aprendizaje automático en la obtención de la cinemática de la marcha humana; como también en las técnicas de aprendizaje profundo para encontrar un patrón biométrico.
- Conocer el funcionamiento del sistema diseñado con los datos adquiridos de los sensores inerciales para una mejorar continua.
- Analizar la aplicación en entorno real con la ayuda de un especialista para obtener resultados factibles para el especialista.
- Evaluar empíricamente los resultados obtenidos en torno a actividades normales.

II. Metodología y Plan de Trabajo

La metodología de investigación en el desarrollo de este trabajo se estructura en las actividades a desarrollarse de acuerdo a los objetivos planteados en la sección anterior usando la estrategia de solapamiento con el objeto de rentabilizar al máximo el tiempo para la ejecución del proyecto (tareas y subtareas), incorporando progresivamente los resultados parciales que se vayan obteniendo de unas a otras. Las tareas y subtareas estan basado para cumplir los objetivos.

A. Tarea 1: Estudio de técnicas y desarrollo de aprendizaje automático en la obtención de la cinemática de la marcha humana; como también en las técnicas de aprendizaje profundo para encontrar un patrón biométrico

- T 1.1 Estudio de la marcha biomecánica de la rodilla y sus respectivas zonas.
- T 1.2 Análisis en revistas indexadas los métodos usados actualmente para detectar inicio/fin de la marcha.
- T 1.3 Análisis en revistas indexadas los métodos usados para el modelamiento de señales.

En esta tarea se prentende conocer el estado actual del problema y las metodologías de inteligencia artificial usadas. Además, ingresar en el mundo de la bio informática para conocer y aplicar términos técnicos y médicos correspondientes a esta área de investigación.

B. Tarea 2: Estudiar el funcionamiento del sistema diseñado con los datos adquiridos de los sensores inerciales para una mejorar continua.

- T 2.1 Recolección de datos del laboratorio de marcha para el análisis del sistema.
- T 2.2 Diseño del sistema y algoritmo de detección
- T 2.3 Probar y adaptar el nuevo un algoritmo inteligente diseñado para la detección de inicio y fin de la marcha humana.

Al concluir esta tarea se pretende desarrollar un sistema de adquisición de datos para recolectar y almacenar set's de datos de la marcha humana con el fin de probar algoritmos inteligentes para el tratamiento de señales y modelado. El fín de esta tarea es que el sistema una vez aprendido a detectar la marcha tenga la capacidad autónoma de obtener el modelo de la marcha para ser almacenado para fines médicos.

C. Tarea 3: Analizar la aplicación en entorno real con la ayuda de un especialista para obtener resultados factibles para el especialista.

- T 3.1 Comparar con otros algoritmos para las pruebas y resultados del algoritmo implementado.
- T 3.2 Diseño del algoritmo de modelado basado en Deep Learning.
- T 3.3 Probar el método de modelado creado para obtener el modelo que será la representación de los datos obtenidos.
- T 3.4 Probar y adaptar el diseño de modelado para comparar con otros algoritmos de inteligencia artificial.

Con esta tarea se pretende usar algoritmos inteligentes que mejore el performance del sistema al ser usado en ambiente real para su experimentación, para lo cual se usará diferentes algoritmos para conocer el mejor performance.

D. Tarea 4: Evaluar empíricamente los resultados obtenidos en torno a actividades normales.

- T 4.1 Probar la aplicación en entorno real con la ayuda de un especialista.
  Una vez realizado las tareas 1 al 3, el fin de esta tarea es usar con el especialista para un fín médico, siendo

uno de los principales usos conocer la afectación de la marcha mediante alguna patología causada para la marcha humana.

III. Relevancia

A. Importancia

El sistema puede ser utilizado como un aparato para la kinesiología para trastornos del movimiento [2], se podría usar para analizar sujetos sanos y evaluación preventiva de la salud por el médico, obteniendo un impacto importante para la sociedad, por ejemplo, problemas de defectos posturales y la marcha de en adultos mayores.

El rápido crecimiento en la proporción de la población mayor de 65 años acelerará los costos de atención médica, aumentando el costo general para la salud pública y el servicio social de un país. Por lo tanto, el diagnóstico científico de la condición postural y estudio de marcha está recibiendo cada vez más atención por parte de la academia de la salud, medicina y deportes en las últimas décadas [2]. Anteriormente se mencionó las ventajas de tener un sistema cuantificable de la marcha como método de prevención, logrando con esto un impacto para el gobierno en ahorro económico en base a la salud pública y servicio social.

Los sistemas WS son nuevas técnicas usadas para el estudio de marcha, debido a que hay industrias que investigan nuevos métodos usando WS con algoritmos inteligentes, el impacto Académico e Industrial será deslumbrante, dado que les interesaría usar las nuevas metodologías encontradas en la investigación.

B. Competencias

Con el proyecto se pretende alcanzar las siguientes competencias.

- Comprensión sistemática de un campo de estudio y dominio de las habilidades
- Métodos de investigación relacionados con dicho campo.
- Capacidad de fomentar, en contextos académicos y profesionales, el avance científico, tecnológico, social, artístico o cultural dentro de una sociedad basada en el conocimiento.
- Desenvolverse en contextos en los que hay poca información específica
- Diseñar, crear, desarrollar y emprender proyectos novedosos e innovadores en su ámbito de conocimiento.
- Trabajar tanto en equipo como de manera autónoma en un contexto internacional o multidisciplinar.
- La crítica y defensa intelectual de soluciones.
- Capacidad de resolución de problemas en entornos nuevos o poco conocidos, aplicando las tecnologías de la Inteligencia Artificial no sólo dentro del ámbito de la Informática, sino también aplicándolas dentro de contextos multidisciplinares.
- Capacidad para razonar en contextos de información imprecisa, incierta o insuficiente.

References

[1] J. Nutt, C. Marsden y P. Thompson, "Human walking and higher-level gait disorders, particularly in the elderly," Neurology, p. 268–268, 1993.

[2] C. Prakash, R. Kumar y N. Mittal, "Recent developments in human gait research: parameters, approaches, applications, machine learning techniques, datasets and challenges," Springer Science+Business Media Dordrecht, pp. 1-40, 2016.

[3] M. Kiss, L. Kocsis y Z. Knoll, "Joint kinematics and spatial-temporal parameters of gait measured by an ultrasound-bases system," Medical Engineering and Physics, vol. 2, pp. 611-620, 2004.

[4] C. Senanayake, "Human assitend tools for gait analysis and Intelligent gait phase detection," Conference on innovative technologies in intelligent systems and industrial applications, 2009.

[5] D. Tarnita, Wearable sensors used for human gait analysis, Romania: Romania Journal of morphology and embryology, 2016.

[6] Y. Song, J. Zhang, L. Cao y M. Sangeux, "On discovering the correlated relationship between static and dynamic data in clinical gait analysis. In: Blockeel H, Kersting K, Nijssen S, Zelezny F (eds) Machine learning and knowledge discovery in databases: European conference, ECML PKDD 2013, Prague," Springer, p. 563–578, 2013.

[7] "Centers for Disease Control and Prevention," Web–based Injury Statistics Query and Reporting System (WISQARS)., p. http://www.cdc.gov/injury/wisqars/index.html., 2017.

[8] G. Owusu, "Ai and computer-based methods in performance evaluation of sporting feats: an overview," Artif Intell Rev 27, p. 57–70, 2007.

[9] B. De Silva, A. Natarajan, M. Motani y K. Chua, "A real-time exercise feedback utility with body sensor networks," 5th international summer school and symposium on medical devices and biosensors, p. 49–52, 2008.

[10] D. Lai, R. Begg y M. Palaniswami, "Computational intelligence in gait research: a perspective on current applications and future challenges," IEEE Trans Inf Technol Biomed, p. 687–702, 2009.

[11] A. Muro, B. Garcia y A. Mendez, "Gait Analysis Methods: An Overview of Wearable and Non-Wearable Systems, Highlighting Clinical Applications," Sensors, vol. 14, pp. 3362-3394, 2014.

[12] H. Liu, Y. Cao y Z. Wang, "Automatic Gait Recognition from a Distance," In Proceedings of Control and Decision Conference (CCDC), pp. 2777-2782, 2010.

[13] M. Derawi, H. Ali y F. Cheikh, "Gait Recognition Using Time-of-Flight Sensor," Researchgate, 2011.

[14] R. Clark, Y. Pua, A. Bryant y M. Hunt, "Validity of the Microsoft Kinect for providing lateral trunk lean feedback during gait retraining. Gait Posture," pp. 1064-1066, 2013.

[15] Z. Xue, D. Ming, W. Song, B. Wan y S. Jin, "Infrared gait recognition based on wavelet transform and support vector machine," Pattern Recognit, pp. 2904-2010, 2010.

[16] J. Zhang, A. Novak y B. Brouwer, "Concurrent validation of Xsens MVN measurement of lower limb joint angular kinematics.," Physiolog Meas, pp. 63-69, 2013.

[17] A. Salarian, P. Burkhard, F. Vingerhoets, B. Jolles y K. Aminian, "Novel approach to reducing number of sensing units for wearable gait analysis systems," IEEE Trans. Biomed. Eng, 2013.

[18] P. Dixon y K. Schutte, "Gait adaptations of older adults on an uneven brick surface can be predicted by age-related physiological changes in strength.," Gait Post 61, p. 257– 262., 2017.

[19] B. Hu, P. Dixon y J. Jacobs, "Machine learning algorithms based on signals from a single wearable inertial sensor can detect surface- and age-related differences in walking," Journal of Biomechanics, 2018.

[20] D.-K. Chew, D. Gouwanda y A. Gopalai, "Investigating running gait using a shoe-integrated wireless inertial sens," TENCON 2015 - 2015 IEEE Region 10 Conference, 2015.

[21] W. Y. "Influence of holding smart phone for acceleration-based gait authentication," Proc. 5th Int. Conf. Emerg. Secur. Technol., pp. 30-33, 2014.

[22] R. Delgado, F. Castro, J. Cozar, M. Marin y N. Guil, "An End-to-End Multi-Task and Fusion CNN for Inertial-Based Gait Recognition," IEEE Access, pp. 1897-1908, 2019.

[23] J. Wang, Y. Chen, S. Hao, X. Peng y L. Hu, Deep learning for sensor-based activity recognition: A Survey, Pattern Recognition Letters, 2018.

[24] S. Tong, Y. Fu, X. Yue y H. Ling, Multi-View Gait Recognition Based on a Spatial-Temporal Deep Neural Network, China: IEEE Acces, 2018.

[25] R. Caldas, M. Mundt, W. Potthast y F. Buarque, A systematic review of gait analysis methods based on inertial sensors and adaptive algorithms, Gait and posture, 2017.

[26] J. Lee, Mellifont R y B. Burkett, "The use of a single inertial sensor to identi stride, step, distance durations of running gait.," Journal of Science d Medicine in Sport 13, pp. 270-273, 2009.

[27] D. Teran, E. Dominguez, "THuman gait model based on a machine learning and filtering noisy signals with recursive algorithm.," 2020 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), Seoul, Korea (South), 2020.

Pseudo-closed elements and closure operators on residuated lattices

Manuel Ojeda-Hernández
Dept. Applied Mathematics
University of Málaga
Málaga, Spain
manuojeda@uma.es

***Abstract*—In this paper the progress done on a PhD thesis is summed up. First, a fuzzy definition of quasi-intents is introduced and its properties are studied. Then, derived from the first topic, a study is done on closure systems in the fuzzy framework, starting with closure systems as crisp sets and then as fuzzy sets. Finally, the one-to-one relation between fuzzy closure systems and closure operators is shown.**

***Index Terms*—Quasi-closed element, fuzzy lattice, closure system, closure operator**

I. Introduction

The main goal of this PhD is, briefly explained, to find the adequate extension of the concepts of Galois connection and closure operator on structures built on relations with values in a residuated lattice and, in particular, to find an extension the notion of pseudo-closed element such that the properties satisfied in crisp lattices are still satisfied and, specifically, the relationship with implication bases, which are key in knowledge representation.

The long term goal is a complete formalization of distinct aspects of imperfect information and the expected advance of this PhD is the development of the mathematical fundamentals for certain problems selected due to their interest according to the recent publications on them.

II. Preliminaries

An algebra $\mathbb{L} = (L, \leq, 0, 1, \to)$ is a complete Heyting algebra, [1], if $(L, \leq)$ is a complete lattice, 0 and 1 are the minimum and maximum elements, respectively, and the following condition holds, for all $p, q, r \in L$

$$p \otimes q \leq r \qquad \text{if and only if} \qquad p \leq q \to r \tag{1}$$

An $\mathbb{L}$-*set* or fuzzy set is a mapping $X\colon U \to L$ from the universe set U to the membership values set L where $X(u)$ means the degree in which u belongs to X. An $\mathbb{L}$-set A is said to be a fuzzy subset of X if $A(u) \leq X(u)$, for all $u \in U$.

As usual, the *core* of X is the set of elements such that $X(x) = 1$ and is denoted by $\mathrm{Core}(X)$.

Finally, a crisp set is considered to be a particular case of $\mathbb{L}$-set by using its characteristic mapping $X\colon U \to \{0, 1\}$ with $X(u) = 1$ iff $u \in X$. Operations with $\mathbb{L}$-sets are defined as usual.

Let $S\colon L^U \times L^U \to L$ be a mapping defined by $S(A, B) = \bigwedge_{x \in U} \big(A(x) \to B(x)\big)$. This mapping is called the subsethood degree relation. Whenever A is a fuzzy subset of B, we have $S(A, B) = 1$, and this is denoted by $A \subseteq B$.

The concepts of fuzzy binary relation, its properties, fuzzy poset and fuzzy lattice are used as in the standard definitions, see [2]–[4]. Throughout this section (A, ρ) will be a fuzzy poset.

The lower and upper cone of a fuzzy set X are defined as

$$X_\rho(a) = \bigwedge_{x \in A} \big(X(x) \to \rho(a, x)\big) \qquad \text{and}$$
$$X^\rho(a) = \bigwedge_{x \in A} \big(X(x) \to \rho(x, a)\big) \qquad \text{for all } a \in A.$$

An element m is said to be a *minimum* (resp. *maximum*) of X if and only if

i) $m \in \mathrm{Core}(X)$ and
ii) $X \subseteq m^\rho$ (resp. $X \subseteq m_\rho$).

If the minimum (resp. maximum) of a set exists then it is unique and is denoted by $\min(X)$ (resp. $\max(X)$).

Similarly, the following concepts were introduced by [5] an element $a \in A$ is an *infimum* (resp. *supremum*) of X if:

i) $a \in \mathrm{Core}(X_\rho)$ (resp. $a \in \mathrm{Core}(X^\rho)$).
ii) $X_\rho \subseteq a_\rho$ (resp. $X^\rho \subseteq a^\rho$).

As a consequence, if $a \in A$ is infimum (resp. supremum) of $X \in L^A$, then $X \subseteq a^\rho$ (resp. $X \subseteq a_\rho$).

As in the classical case, if the infimum (resp. supremum) of a set exists then it is unique and is denoted by $\bigsqcap X$ (resp. $\bigsqcup X$). In addition, an element $m \in A$ is a *minimum* (resp. *maximum*) of X if and only if m is an infimum (resp. supremum) of X and $X(m) = 1$.

Theorem 1: Let $X \in L^A$. An element $a \in A$ is infimum (resp. supremum) of X if and only if $X_\rho = a_\rho$ (resp. $X^\rho = a^\rho$).

A complete fuzzy lattice is a set such that every fuzzy subset $X \in L^A$ has supremum and infimum.

We denote by $\bot$ and $\top$ the infimum and the supremum of the complete fuzzy lattice, respectively. The pair (L^U, S) is an example of complete fuzzy lattice [6], which is called the *L-powerset lattice of* U.

The product and residuum of $\mathbb{L}$ have been extended to an external operation on the powerset lattice as follows: for all $a, x \in L$, $A \in L^U$,

$$\begin{aligned}(a \otimes A)(x) &= a \otimes A(x) \text{ and} \\ (a \to A)(x) &= a \to A(x).\end{aligned} \tag{2}$$

Notice that these operations involve elements of L and elements of L^U, hence they are particularly defined on the powerset lattice.

The focus of this paper is on closure operators and closure systems. The definition of closure operator is a direct extension of the classical concept since it is a mapping which satisfies the fuzzy extensions of isotony, inflationarity and idempotency whereas there is not a unique extension of the notion of closure system to a fuzzy setting. The most referenced one was given by Bělohlávek in [7].

III. Quasi-closed elements

In the literature, there are two main ways of defining pseudo-closed elements. On the one hand, there is one which depends on a recursive definition of pseudo-closedness, making the computation of these elements hard. However, an extension of this definition to the fuzzy framework was done in [8]. On the other hand, the second definition depends on the concept of quasi-closed element. This is the direction we followed in [4].

In the crisp case there are five properties which are equivalent to being quasi-closed. After comparing all the immediate extensions to fuzzy framework of them we chose the most general one.

Definition 2: Given a closure operator $\mathtt{c}$ on a fuzzy poset (A, ρ), an element $q \in A$ is said to be *quasi-closed* (with respect to $\mathtt{c}$) if, for all $a \in A$,

$$\rho(a,q) \otimes \neg\rho(q,a) \otimes \rho(\mathtt{c}(a),\mathtt{c}(q)) \otimes \neg\rho(\mathtt{c}(q),\mathtt{c}(a)) \leq \rho(\mathtt{c}(a),q).$$

In order to ease the reading we will use the following notation.

Notation 1: Given a closure operator $\mathtt{c}$ on a fuzzy poset (A, ρ) and $q \in A$, we use X_q to denote the $\mathbb{L}$-set with membership function defined as follows, for each $a \in A$

$$X_q(a) = \rho(a,q) \otimes \neg\rho(q,a) \otimes \rho(\mathtt{c}(a),\mathtt{c}(q)) \otimes \neg\rho(\mathtt{c}(q),\mathtt{c}(a)).$$

With this notation, an element q is quasi-closed iff $X_q(a) \leq \rho(\mathtt{c}(a), q)$, for all $a \in A$. Notice that the definition of quasi-closedness is crisp, that is, elements are either quasi-closed or not. The next step was to find a fuzzy definition of quasi-closed element which gives the degree to which an element is quasi-closed. The definition we found is the following.

Definition 3: Given a closure operator $\mathtt{c}$ on a fuzzy poset (A, ρ), for any $q \in A$, we define the *degree in which q is quasi-closed* as follows

$$QC(q) = \bigwedge_{x \in A} \left[X_q(x) \to \rho(\mathtt{c}(x), q) \right].$$

The two definitions of quasi-closedness, the crisp one and the graded one just presented, fit. Specifically, if we consider QC as a fuzzy set, its 1-cut is precisely the set of the quasi-closed elements.

Theorem 4: Let $\mathtt{c}$ be a closure operator on a fuzzy poset (A, ρ) and $q \in A$. Then, $QC(q) = 1$ if and only if q is quasi-closed.

In the classical setting, there is an if-and-only-if condition for a set to be quasi-closed based on an operator that is usually denoted by $^\circ$. We can do an analogous characterization here, even in the graded setting. The definition of this operator depends on the existence of suprema so we need (A, ρ) to be a complete fuzzy lattice.

Definition 5: Let $\mathtt{c}$ be a closure operator on a complete fuzzy lattice (A, ρ) and $q \in A$. We define the element q° as follows:

$$q^\circ = q \sqcup \bigsqcup \mathtt{c}(X_q).$$

This is not a quasi-closed element in general, not even in the crisp case. However, this definition of $^\circ$ was considered adequate because it satisfies the next theorem.

Theorem 6: Let $\mathtt{c}$ be a closure operator on a complete fuzzy lattice (A, ρ). Then, $QC(q) = \rho(q^\circ, q)$, for all $q \in A$.

The set of quasi-closed elements is a closure system in the classical case, associated with a closure operator called the saturation operator. In order to check if this holds with the definition given here we found a new problem, a definition of closure system in the fuzzy setting without making use of the powerset lattice structure. The most cited one we found is the one by Bělohlávek.

Definition 7 ([7]): Let K be a filter in L. A system $\mathcal{F} = \{A_i \in L^X \mid i \in I\}$ is called closed under S_K-intersections iff for each $A \in L^U$ it holds that

$$\bigcap_{\substack{i \in I \\ S(A,A_i) \in K}} (S(A, A_i) \to A_i) \in \mathcal{F},$$

where the external operation introduced in (2) is used. A system closed under S_K-intersections will be called an L_K-closure system. For $K = L$ the subscript will be omitted.

Notice that in Bělohlávek's definition, fuzzy closure systems are crisp sets.

The advances on closure systems avoiding the explicit use of powerset structure is done in the next section.

IV. Closure systems

As in the classical case, there is a bijection between closure operators and closure systems. We can find different definitions of closure systems depending on the ordered structure on which the fuzzy closure operator is defined. Thus, a fuzzy closure system has been defined on L-ordered sets [9], on fuzzy preposets [2], and on fuzzy preordered structures [10], on fuzzy sublattices [11]. The closest definition to the approach used in this paper is the one by Bělohlávek on L-powerset lattices in [7].

We have revisited a definition of (fuzzy) closure system in the framework of complete fuzzy lattices. This definition is inspired on the notion of closure system in the case of classical complete lattices, where they are $\wedge$-subsemilattices and was introduced by Bělohlávek in [12].

Definition 8: Let (A, ρ) be a fuzzy complete lattice. A crisp subset $\mathcal{F} \subseteq A$ is said to be a closure system if $\bigsqcap X \in \mathcal{F}$ for any fuzzy subset X of $\mathcal{F}$.

Notice that this definition does not use explicit use of the powerset lattice structure. This definition is well-behaved, as it maintains the one-to-one relation with closure operators and, in addition, when applied in a powerset lattice it is equivalent to Definition 7.

The next step was to consider closure systems as fuzzy sets. There have been several attempts to extend closure systems to fuzzy sets, for instance we have L-closure L-systems, proposed by Bělohlávek in [7] or the one by Liu and Lu in [13].

Definition 9: Let (A, ρ) be a complete fuzzy lattice. We say that a fuzzy set $\Phi \in L^A$ is a fuzzy closure system if $\mathrm{Core}(\Phi)$ is a closure system and, for all $x, y \in A$, the following condition holds

$$\Phi(x) \otimes (x \approx_{\otimes} y) \leq \Phi(y).$$

This additional property is commonly known as extensionality of Φ or compatibility of Φ with $\approx_{\otimes}$.

From the results on crisp closure systems and extensionality we get the following.

Corollary 10: Let $\mathbb{A} = (A, \rho)$ be a complete fuzzy lattice. The following assertions hold:

i) If Φ is a fuzzy closure system on $\mathbb{A}$, the mapping $\mathsf{c}_\Phi \colon A \to A$ defined as $\mathsf{c}_\Phi(a) = \bigcap(a^\rho \otimes \Phi)$ is a closure operator on $\mathbb{A}$.
ii) If c is a closure operator on $\mathbb{A}$, the fuzzy set Φ_c defined as $\Phi_\mathsf{c}(a) = \rho(\mathsf{c}(a), a)$ is a fuzzy closure system on $\mathbb{A}$.
iii) If Φ is a fuzzy closure system on $\mathbb{A}$, then $\Phi = \Phi_{\mathsf{c}_\Phi}$.
iv) If $\mathsf{c} \colon A \to A$ is a closure operator on $\mathbb{A}$, then $\mathsf{c}_{\Phi_\mathsf{c}} = \mathsf{c}$.

V. Conclusions and future work

As a prospect of future work we plan to extend the definition of fuzzy closure operator, and its corresponding closure system, to relational structures as an extension of the results presented in [14]. On the other hand, the results obtained will allow us to advance in the research initiated in [4] on the concepts of quasi-closed and pseudo-closed elements.

Acknowledgmet

I would like to thank Vylem Vychodil, who originally posed the problem of fuzzy pseudo-intents that lead us in this direction. This work has been partially supported by the FPU19/01467 internship (Spanish MCIU) and the Spanish research projects with reference TIN2017-89023-P (MCIU/AEI/FEDER), PGC2018-095869-B-I00 and UMA2018-FEDERJA-001 (JA/UMA/FEDER).

References

[1] P. Hájek, *Metamathematics of Fuzzy Logic*, ser. Trends in Logic. Springer Netherlands, 2013.
[2] I. P. Cabrera, P. Cordero, F. García-Pardo, M. Ojeda-Aciego, and B. De Baets, "On the construction of adjunctions between a fuzzy preposet and an unstructured set," *Fuzzy Sets and Systems*, vol. 320, pp. 81–92, 2017.
[3] W. Yao and L. X. Lu, "Fuzzy Galois connections on fuzzy posets," *Mathematical Logic Quarterly*, vol. 55, pp. 105–112, 2009.
[4] M. Ojeda-Hernández, I. P. Cabrera, and P. Cordero, "Quasi-closed elements in fuzzy posets," *Journal of Computational and Applied Mathematics*, vol. In Press, pp. 113–390, 2021.
[5] R. Bělohlávek, *Fuzzy Relational Systems*. Springer, 2002.
[6] R. Bělohlávek, "Lattice-type fuzzy order is uniquely given by its 1-cut: proof and consequences," *Fuzzy Sets and Systems*, vol. 143, no. 3, pp. 447–458, 2004.
[7] R. Bělohlávek, "Fuzzy closure operators," *Journal of Mathematical Analysis and Applications*, vol. 262, pp. 473–489, 2001.
[8] V. Vychodil, "Computing sets of graded attribute implications with witnessed non-redundancy," *Information Sciences*, vol. 351, pp. 90 – 100, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0020025516301451
[9] L. Guo, Q. Zhang, and Q. Li, "Fuzzy closure systems on L-ordered sets," *Mathematical Logic Quarterly*, vol. 57, no. 3, pp. 281–291, 2011.
[10] I. P. Cabrera, P. Cordero, F. García-Pardo, M. Ojeda-Aciego, and B. De Baets, "Galois connections between a fuzzy preordered structure and a general fuzzy structure," *IEEE Trans. Fuzzy Systems*, vol. 26, no. 3, pp. 1274–1287, 2018.
[11] B. Šešelja and A. Tepavčević, "Fuzzifying closure systems and fuzzy lattices," *Lecture Notes in Computer Sciences*, vol. 4482, pp. 111–118, 2007.
[12] R. Bělohlávek, "Lattice type fuzzy order and closure operators in fuzzy ordered sets," in *Proceedings Joint 9th IFSA World Congress and 20th NAFIPS International Conference*, vol. 4, 2001, pp. 2281–2286 vol.4.
[13] Y.-H. Liu and L.-X. Lu, "L-closure operators, l-closure systems and l-closure l-systems on complete l-ordered sets," in *2010 Seventh International Conference on Fuzzy Systems and Knowledge Discovery*, vol. 1, 2010, pp. 216–218.
[14] I. P. Cabrera, P. Cordero, E. Muñoz-Velasco, M. Ojeda-Aciego, and B. D. Baets, "Relational Galois connections between transitive digraphs: Characterization and construction," *Inf. Sci.*, vol. 519, pp. 439–450, 2020.

Looking for a Synergy between Boltzmann Machines and Markov Networks

Arkaitz Bidaurrazaga[1,2]
[1]*Basque Center for Applied Mathematics (BCAM)*
[2]*University of the Basque Country UPV/EHU*
Bilbao, Spain
abidaurrazaga@bcamath.org

Aritz Pérez[1,2,3], Roberto Santana[2,3]
[3]*Intelligent System Group(ISG)*
San Sebastian, Spain
aperez@bcamath.org,roberto.santana@ehu.eus

Starting date: 1 July 2021

***Abstract*—Energy-based models (EBM) represent probability distributions that factorize according to an additively decomposable energy-function. Depending on the form of the additive decomposition we can define different families of EBMs. The main research line of this thesis consists of analyzing and finding synergies between Boltzmann Machines (BM) and Markov Networks (MN), and propose the Markov-Boltzmann models (MBM), a more flexible EBM that generalize them. On the one hand, BMs decomposes the energy into all the factors of order one, i.e., all pairs of variables. On the other hand, MNs decomposes into higher order factors determined by an undirected graph from which it is possible to read conditional independencies. MBMs factorize according to an arbitrary undirected graph while controlling the maximum order of the factors. Given an undirected graph, MBMs will allow us to improve the efficiency of the learning algorithms of BMs, and produce a more compact representation of probability distributions than MNs. In addition, during the thesis we will develop structural learning algorithms for MBM, and we will deal with machine learning problems such as supervised classification.**

***Index Terms*—**

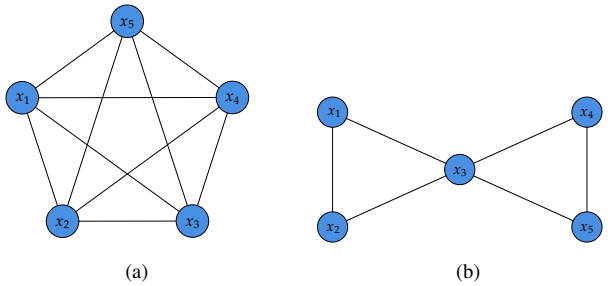
I. Introduction

EBMs assign an energy $E(\mathbf{x}, \theta)$ to each instance $\mathbf{x}$ of dimension n, where θ represents the parameters of the energy-function. Using the energy function they define the probability distribution over $\mathbf{X}$ by assigning a probability proportional to the exponential of the negative energy of each instance $\mathbf{x}$, i.e., $q(\mathbf{x}, \theta) = \frac{e^{-E(\mathbf{x},\theta)}}{Z(\theta)}$, where $Z(\theta)$ is the partition function and depends on the parameter θ. The partition function $Z(\theta) = \sum_{\mathbf{x}\in\mathcal{X}} e^{-E(\mathbf{x},\theta)}$ ensures the distribution to be normalized, being $\mathcal{X}$ the suport of the variable $\mathbf{X}$, i.e, the set of all possible values that $\mathbf{x}$ can take. Unfortunately, as n increases the size of $\mathcal{X}$ grows exponentially and the computation of the partition function becomes intractable. In these situations, the partition function has to be approximated using numerical methods, which makes parametric learning of EBMs for general energy functions difficult.

The main goal of the EBM training phase is to learn an unknown probability distribution, using instances sampled from that distribution, and computing the parameters that optimize the approximation $q(\mathbf{x}, \theta)$ according a score function . Usually, the parametric learning approaches try to minimize the Kullback-Leibler divergence (KL divergence) between the empirical distribution and the obtained model, $\min_\theta D_{KL}(\hat{p}(\mathbf{x})|q(\mathbf{x}, \theta))$.

In this thesis we are interested in EBMs that additively decompose the energy-function as $E(\mathbf{x}) = \sum_{R\in\mathcal{R}} f_R(\mathbf{x}_R)$, where R is a subset of indices associated to variables with $\mathbf{x}_R = (x_i : i \in R)$, f_R is a non-negative function that depends on $\mathbf{x}_R$, and $\mathcal{R}$ is a collection of sets of indices that qualitatively describes the decomposition. The additive decomposition given by $\mathcal{R}$ can be alternatively viewed as the factorization of the distribution represented by the EBM as $q(\mathbf{x}) = \prod_{R\in\mathcal{R}} \phi_R(\mathbf{x}_R)$, where $\phi_R(\mathbf{x}_R)$ are non-negative functions called the potentials.

Firstly, we will focus on two EBMs: Boltzmann Machines (BM) and Markov Networks (MN). These two models differ by how they factorize the probability distribution. BMs consider all the first order potentials, that is, potentials over pairs of variables. Thus for BMs, $\mathcal{R}$ corresponds to the set of all pairs of indices, i.e., $\mathcal{R} = \{(i, j) : 1 \leq i < j \leq n\}$. This allows us to approximate general probability distributions with a polynomial number of parameters $\mathcal{O}(n^2)$. More specifically, BM's energy-function is defined as $E(\mathbf{x}, \theta) = \sum_{i<j} x_i w_{i,j} x_j + \sum_i b_i x_i$ where $\theta = \{w_{i,j}, b_i | i, j \in \{1, \ldots, n\}\}$ are the weights of the interactions and biases, respectively. This definition concurs with the previously established factorization of the probability. The weights describe the first order interactions between variables x_i and x_j. It can be seen from Figure 1 that each edge would correspond to a weight describing the interaction between two variables. In [1], higher order BMs are reviewed, where a k-order BM is defined as a BM, but each R corresponds to a tuple of



(a) (b)

Fig. 1: (a) A graph representing a Boltzmann machine with 5 variables. Here the probability factorizes as: $p(x_1, x_2, x_3, x_4, x_5) = \phi_{1,2}(x_1, x_2)\phi_{1,3}(x_1, x_3)...\phi_{4,5}(x_4, x_5)$ for all pair of variables $x_{i,j}$ with $i < j$. (b) This graph represents a Markov Network, where there are two cliques $\{x_1, x_2, x_3\}$ and $\{x_3, x_4, x_5\}$. Hence, the probability factorizes as $p(x_1, x_2, x_3, x_4, x_5) = \phi_{1,2,3}(x_1, x_2, x_3)\phi_{3,4,5}(x_3, x_4, x_5)$.

size $k+1$. These models have well defined training methods, which are a generalization of the methods for first order BMs. However, they are not used in practice because the number of parameters to learn is $\mathcal{O}(n^{k+1})$ and n is typically large. On the other hand, MNs factorize the distribution according to an undirected graph (UG), and thus the factorization changes depending on the graph considered. In MNs, each potential is defined over a maximal clique[1] of the UG. For MNs, $\mathcal{R} = \{R : R \text{ is a maximal clique in } G\}$, where G is the corresponding UG. From here on we will call to $\mathcal{R}$ the structure of the EBM. In Figure 1, We show the two graphs representing BMs and MNs. Note that, while we are using two UGs, their semantic is different. In this example, assuming that the variables are binary, the number of parameters of the BM is 5^2 while for the MN is $2^3 + 2^3$.

MNs are more flexible than BMs, because they can control the number of parameters to be modeled, that are exponential in the size of the largest maximal clique. On the other hand, BMs does not introduce any conditional independence because they consider to model all the potentials of order one. In this work, we seek to improve these models by combining their main attributes. We shall find a model that factorizes according to an UG without requiring to model the exponential number of parameters required by the MN.

Controlling the number of parameters while obtaining an expressive distribution is one of the main motivations for our work. We will analyze and develop a new model named Markov-Boltzmann model (MBM), which factorizes according to an UG into potentials with a controlled number of parameters. Such that we can model the same conditional dependencies as a MN with less parameters. In other words, probability distributions can be represented in a compact and flexible manner. For a MBM, we read the UG in the same manner as a MN. Nonetheless, each factor of a clique will factorize into $k+1$-sized tuples of variables, when the size of the clique is bigger than $k+1$. Thus $p(\mathbf{x}; G, k) = \prod_{R\in\mathcal{R}} \phi_R(\mathbf{x}_R)$, where $\phi_R(\mathbf{x}_R) = \prod_{S\subset R;|S|=k+1} \phi_S(\mathbf{x}_S)$ for $|R| > k+1$. The parameter k can be set by the user, in this manner, the user will have control on the number of parameters to be trained. The number of parameters is $\mathcal{O}(c^{k+1})$, where c is the size of the biggest clique in G. Following the example in Figure 1, the factorization of a MBM of first order with the graph (b) is $\phi_{1,2,3}(x_1, x_2, x_3) = \phi_{1,2}(x_1, x_2)\phi_{1,3}(x_1, x_3)\phi_{2,3}(x_2, x_3)$ for the first factor and $\phi_{3,4,5}(x_3, x_4, x_5) = \phi_{3,4}(x_3, x_4)\phi_{3,5}(x_3, x_5)\phi_{4,5}(x_4, x_5)$ for the second factor. This will be the basis for the development of parametric and structural learning algorithms for MBM.

Moreover, the energy-function can be extended by adding hidden or latent variables $\mathbf{h}$. The Restricted Boltzmann Machines (RBM) are defined as an extension of BMs, with an energy-function $E(\mathbf{x}, \mathbf{h}, \theta) = \sum_{i,j} x_i w_{i,j} h_j$, where $j \in \{1, \ldots, m\}$, being m the number of hidden states, and for the sake of simplicity we have suppressed the bias terms. These models are called restricted, because interactions between visible states ($\mathbf{x}$) are not permitted, as well as interactions between hidden states; this makes sampling more efficient using MCMC methods. RBMs are more widely used in the literature, and thus, combining them with MNs will be the next objective after verifying the effectiveness of MBMs.

In order to adjust the parameters of EBMs, the Stochastic Gradient Descent (SGD) method is used, which was first introduced in [2]. Training these models with SGD requires computing averages with respect to the models distribution. However, as we have mentioned before, the model distribution is unnormalized, so it is not possible to sample directly from q. Therefore, the expected value is approximated obtaining samples from the model with Markov Chain Monte Carlo (MCMC) methods. In [3], a review on how to train RBMs is conducted (in the literature RBMs are more used than BMs, because of their higher flexibility thanks to the hidden states), where the authors also specify different methods to approximate or avoid computing the partition function. In the literature, it is common to maximize the LogLikehood (LL) of a model's distribution $q(\mathbf{x}, \theta)$. Maximizing the LL of $q(\mathbf{x}, \theta)$ is equivalent to minimizing the Kullback-Leibler divergence (KL divergence) between the experimental distribution $\hat{p}(\mathbf{x})$ and $q(\mathbf{x}, \theta)$ [4]. Therefore, these two metrics were used to compare performance. However, many alternative metrics have been proposed. In [5], Montavon et al. propose to use the Wasserstein distance instead of KL divergence. The authors in [6] define a family of distances between two probability distributions which are called α-divergences, where the KL divergence is a special case for $\alpha = 1$. Regarding MBMs, exploiting their properties in order to optimize minimizing certains scores will be a matter of interest, such as employing their factorization properties so that parallel learning is possible.

[1]A clique of a UG is any complete subset, and a maximal clique is any clique which is not a proper subset of another clique.

II. Objectives

The main objective is to develop structural and parametric learning methods for MBMs. First, we will analyze the synergies between BMs and MNs which allow to improve the versatility of these models and the design of efficient learning procedures, in order to represent accurate approximations of probability distributions. This goal can be broken-down into the following sub-objectives:

1) **Study the advantages of training first order MBMs.** First order MBMs are BMs with nullified weights. When the analysis of the data indicates that two variables x_i, x_j are conditionally independent, or we know this by prior knowledge, the independence should be learned by the model, for BMs it translates into having $w_{i,j} = 0$. Hence, we assume that setting weights of a BM to 0, for pairs of variables that are known to be conditionally independent, should be better than initializing them at random.
2) **Parametric learning algorithms for k-order MBMs.** The main idea of seeking a synergy between BMs and MNs, is to construct a model that has the expressiveness of a MN but which requires less number of parameters like a BM. Assuming we are able to compute the interactions between variables, without explicitly training an EBM; we may be able to set a maximum size c for cliques, which determines the number of parameters in the k-order MBMs (c^{k+1}). In this manner, we may be able to obtain good probability approximations without the cost of expensive computation.
3) **Extend to k-order Restricted MBM.** RBMs are an extension of BMs, where latent variables help the representavility of the distribution. Hence, developing a k-order MBM using latent variables is the most convenient step in order to further improve our model. We call this model the k-order Restricted MBM (RMBM). The main challenge for this model will be to translate the conditional dependencies between variables, given the corresponding graph, into the interactions between visible and hidden variables.
4) **Analyze different training methods for EBMs.** The most common training method for EBMs is based on minimizing the KL divergence with gradient descent. Nonetheless, a well-known issue with EBMs complicates the computation of the gradient, which is computing/approximating the partition function $Z(\theta)$. When the search space of our problem is big enough, computing $Z(\theta)$ is intractable, which is necessary to compute the gradient of the KL divergence because it requires to average over the model's distribution $q(\mathbf{x};\theta)$. This average is usually approximated via Gibbs Sampling (GS) [7], which is a MCMC method that generates samples from the model, in order to approximate the average over $q(\mathbf{x};\theta)$. However, this method is not efficient for high-dimensional data because many iterations of sampling are needed in order to obtain a good approximation, and this is a high time consuming learning algorithm. Recent alternatives to GS are reviewed in [4], [8]. In this thesis we will explore how these methods could help to improve training MBMs.
5) **Optimize parameter initialization for MBMs.** When we train a ML model, using a proper initialization of the model parameters is difficult [9]. If a pre-trained model exists which tackles the type of problem we are interested in, we can fine-tune the model with our dataset. According to [10], this will probably yield to a faster and better training compared to a randomly initialized model. However, whenever we face a novel problem, or whether we are experimenting with a different architecture, we are forced to initialize parameters randomly. In [9], authors also propose some initialization techniques depending on the number of inputs n and outputs m the neural networks have. In this thesis, we want to show that we can do better than initializing randomly. We aim to define an initialization strategy that does not rely in pure randomness, and which exploits the information contained in the dataset used for training.

III. Methodology and Work Plan

In this section, a general overview of the proposed methodology is presented. The work plan is defined in order to fulfill each one of the objectives of this thesis. The preliminary findings may be presented in conferences, preferably international. More concretely, the methodology for each objective is defined as follows:

1) **Study the advantages of training first order MBMs:**
   a) In-depth revision of EBMS and possible usage of structural/independence information when training.
   b) Conduct experiments with small artificial datasets, for which we set the structure of dependencies among variables. Exploit the prior knowledge when training first order MBMs, and observe whether it improves the results versus a regular BM.
   c) Extend these experiments to bigger and real datasets, presumedly to image datasets, where we can assume that only nearest pixels are dependent of each other.
   d) Write a paper with the obtained results and conclusions. Submit it to an international conference.
2) **Parametric learning algorithms for k-order MBMs:**
   a) Thorough research on the state-of-the-art methods to implement higher order BMs. Search for training methods that make sampling possible or which do not rely on sampling.
   b) Propose a novel method to train k-order MBMs, being aware that there will be a trade-off between performance and computational cost.
   c) Validate the method using artificial and real datasets.
   d) Write a paper and submit it to an international conference.

3) **Extend to k-order Restricted MBM:**
   a) In-depth research on the literature of RBMs; how to train them, alternative architectures, etc.
   b) Generalize the k-order MBM using latent variables, inspired by RBMs.
   c) Validate the method using artificial and real data.
   d) Compare k-order RMBM with k-order MBM.
   e) Write a review of metrics used to compare models, and submission to a JCR journal.
4) **Analyze different training methods for EBMs:**
   a) Revision of training methods for EBMs, with emphasis on those that propose alternatives to GS. New sampling strategies, more efficient than GS, will also be of our interest.
   b) Since at the time of addressing this task, we would have decided a proper metric, compare selected training methods, and choose those that fit in our criteria, always being aware of the trade-off between training elapsed time and performance.
   c) Write a paper with results and conclusions, and submit it to an international conference.
5) **Optimize parameter initialization for MBMs:**
   a) Try to exploit the properties of different metrics in order to optimize training our model.
   b) Explore how random initialization performs. Then analyze the converged parameters, in order to extract patterns.
   c) If a pattern is found, propose methods to induce this pattern early in the initialization step, such that it helps speeding up training.
   d) Execute trial and error experiments with different initialization methods that we propose, and compare them with the random initialization.
   e) Compare this initialization method using a bigger or real dataset. And conclude whether it is better than initializing randomly.
   f) Write a paper with results and conclusions, and submit it to a JCR journal.

IV. Current work

We have already fulfilled points a) and b) of the first sub-objective, and points a), b) and c) of the fourth sub-objective, and we have observed that:

1) **BM and first order MBMs converge to similar weights.** As we have mentioned before, if two variables x_i, x_j are conditionally independent then the model should learn this independence. Translating this to BMs means that the interaction weight $w_{i,j}$ should be 0. Thus, we can force those weights to be 0 when the independence is known. We have conducted some experiments with artificial datasets.
   We have generated different datasets with n binary variables, where samples were obtained from a data generator establishing which variables were independent via an UG. We run experiments for $n \in \{8, 9, 10, 11, 12, 13\}$, and trained two BMs, one where the parameters $w_{i,j}$ were set to 0 whenever x_i, x_j were independent (by our choice) and another model with every parameter free to train.
   In Figure 2 we show a heatmap representation of the converged weight matrices for every n, after 200 epochs of training. The number of training samples was $N\_train = 2^{13}$. The scale determines that the darkest regions are related to values of $w_{i,j}$ closest to zero. In the MBM model, the black boxes that are above and below the diagonal are the parameters that were set to 0, and which represent the conditional independence between variables. Notice how the BM converges to a very similar configuration, respecting the 0-blocks on the MBM. Hence, we can assume that inserting prior knowledge of the distribution factorization helps training BMs, and thus a first order MBM is more suitable than a regular BM.
   Nonetheless, we must conduct point c), because these conclusions were driven from small and artificial datasets. There still exists the question of the effectiveness of this method for datasets with a large number of variables. As well as how to apply this approach in datasets where we are not aware of the underlying factorization of the real distribution.
2) **An approach for initializing first order MBMs.** Analyzing how the weight matrices evolve through training, we have computed the correlation between the matrix elements and some metrics that describe the dependence between two variables. First of all, we computed the mutual information (MI) for all pairs of x_i, x_j, and computed the correlation between the vectors $MI(x_i, x_j)$ and $|w_{i,j}|$, where the elements of each vector are ordered in the same way and satisfying $i < j$. The absolute value of the weights were used, because MI takes values in the range $[0, 1]$, thus, considering negative values would obviously lead to a low correlation. We observed that the weight matrix (in absolute values) when converged is highly correlated with the MI. Consequently, we proposed initializing the weight matrix with the values of MI.
   Nonetheless, this approach raises a clear problem, which is that there will be no negative weights when initializing $w_{i,j}$. The weight matrix needs freedom to take negative values as well, in Figure 2 we can see that many elements are indeed negative. Hence, we needed a metric that relates the interaction between two variables which can take negative values. In [11], Kendall proposed a rank based correlation, which ranges between $[-1, 1]$ and is known as Kendall tau correlation.
   We defined two sets of elements of $w_{i,j}$ in our experiments, R_1 is the set of elements that are free for the free and structured model, and R_0 is the set of elements that are forced to be 0 in the structured model. We have conducted experiments initializing randomly both models, in Figure 3 we show the evolution of the

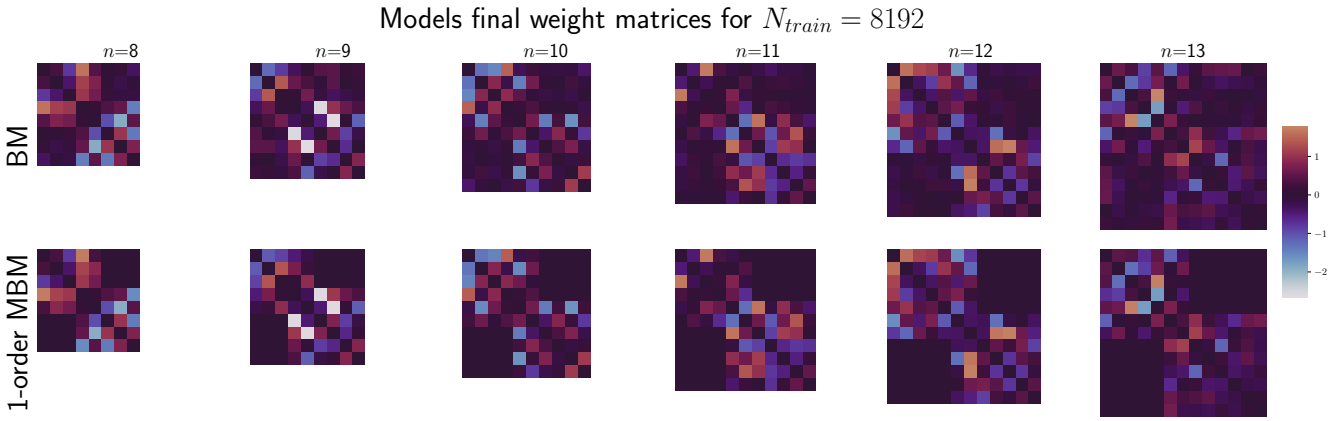


Fig. 2: After 200 epochs of training with a train set size of 2^{13}, the converged weights matrices are represented in this figure as heatmaps. Every matrix was initialized randomly, and we compare the results for different values of n. For the elements which are forced to be 0 in the MBM, notice how the BM reaches values very close to zero for those elements. This demonstrates that the BM learns the conditional independence between those variables, which proves our assumption that inserting structural priors can improve training, since having some parameters set to 0 during the training process translates into less parameters to be adjusted.

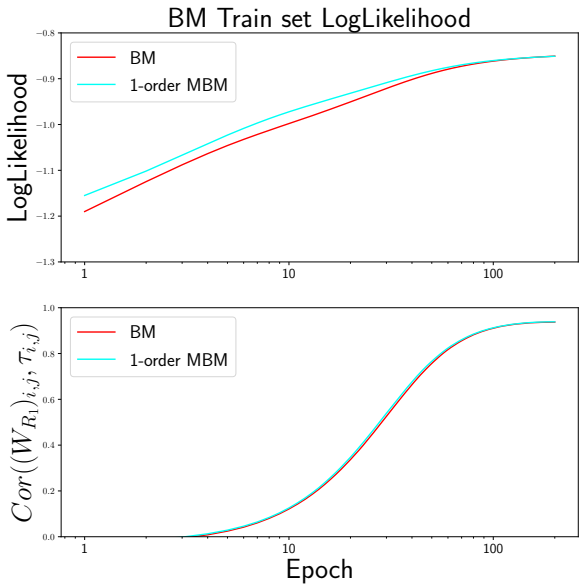


Fig. 3: BM and first order MBM training along 200 epochs with randomly initialized weights, for $n = 12$ binary variables and $N_{train} = 2^{13}$. The Loglikelihood is normalized with the number of variables n and size of the training set N_train, in order to obtain comparable values for future analysis with different values of n. In the second graph, the Pearson correlation is computed between the Kendall tau and the elements of $w_{i,j}$ in R_1, that is, the elements that are not forced to be 0 in the MBM.

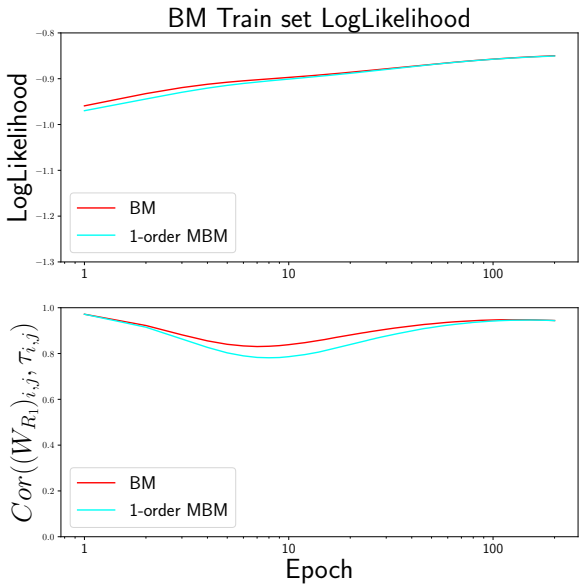


Fig. 4: BM and first order MBM training along 200 epochs, with weights initialized to $\tau_{i,j}$, for $n = 12$ binary variables and $N_{train} = 2^{13}$. The Loglikelihood is normalized with the number of variables n and size of the training set N_train, in order to obtain comparable values for future analysis with different values of n. In the second graph, the Pearson correlation is computed between the Kendall tau and the elements of $w_{i,j}$ in R_1, that is, the elements that are not forced to be 0 in the MBM.

Loglikelihood of the train set, as well as the correlation between the elements of the weight matrix in R_1 with respect to the metric Kendall tau $\tau_{i,j}$. At the end of the training we see that they are highly correlated. This is of great interest because we are considering the actual value of the parameters, instead of their absolute values. Observing that $w_{i,j}$ is highly correlated with Kendall tau at convergence, and that the parameters order is of $\mathcal{O}(1)$ (see Figure 2), we expected initializing with Kendall tau to be a good choice. In Figure 4 we conducted the same experiment but with this new initialization. As anticipated the weight matrix (R_1 set) starts highly correlated, then its correlation with τ decreases until convergence where it reaches a high correlation again. This small variance of the correlation can be due to some parameters having very similar values, thus small changes could swap their order resulting in this variance. Notice that there is a substantial improvement in the initial Loglikelihood (≈ -0.98), comparing with Figure 3 where the weights were randomly initialized (≈ -1.18).

V. Relevance

We think that an insight view of how MBMs work can help understanding more complex EBMs. RBMs are widely used in the literature, and the extension of MBMs to RBMs might turn into even more expressive distributions. RBMs have been used to diagnose Alzheimer's disease [12], and to make facial expression recognition from thermal infrared images [13] as well. BMs and RBMs are good generative models, they can generate images and even sequential data as described in [14], [15]. Thus, imposing a sensible conditional graph to a set of variables, and using a MBM, can speed up training, because less parameters need to be adjusted, and the learning process can be paralelized by training groups of parameters separated by the cliques they define in the graph.

There are many challenges in the Deep Learning field [16], and one of them is to train faster state-of-the-art models. Thus, improving the training step of EBMs is a matter of interest, since nowadays problems involve larger and larger datasets. Scalable training algorithms will be needed to train models with so many instances. One of the most inefficient steps when training an EBM is GS, because it scales badly as the number of dimensions and the size of the dataset increase. Hence, alternative sampling approaches or other training techniques that do not require sampling should be explored. Furthermore, when training, the number of epochs to reach convergence is crucial. Thus, good initialization techniques that can converge faster should also be investigated. We think we are taking good steps as showcased in the Current Work section. Finally, a better representative capability of the data distribution can be achieved by training MBMs with higher order interactions. This implies computationally more expensive procedures, so methods that have a good balance between representation accuracy and complexity will be studied.

Acknowledgment

This research is supported by the Spanish Ministry of Science and Innovation through BCAM Severo Ochoa excellence accreditation SEV-2017-0718. R. Santana acknowledges support by the Basque Government (ELKARTEK 3KIA (KK2020/00049), IT1244-19), and by the Spanish Ministry of Economy and Competitiveness MINECO (project TIN2016-78365-R and PID2019-104966GB-I00).

References

[1] T. J. Sejnowski, "Higher-order Boltzmann machines," in *AIP Conference Proceedings*, vol. 151, pp. 398–403, American Institute of Physics, 1986.

[2] J. Kiefer, J. Wolfowitz, *et al.*, "Stochastic estimation of the maximum of a regression function," *The Annals of Mathematical Statistics*, vol. 23, no. 3, pp. 462–466, 1952.

[3] G. Montúfar, "Restricted Boltzmann machines: Introduction and review," in *Information Geometry and Its Applications IV*, pp. 75–115, Springer, 2016.

[4] Y. Song and D. P. Kingma, "How to Train Your Energy-Based Models," *arXiv preprint arXiv:2101.03288*, 2021.

[5] G. Montavon, K.-R. Müller, and M. Cuturi, "Wasserstein training of restricted Boltzmann machines," in *Proceedings of the 30th International Conference on Neural Information Processing Systems*, pp. 3718–3726, 2016.

[6] T. Minka, "Divergence measures and message passing," Tech. Rep. MSR-TR-2005-173, 2005.

[7] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 6, pp. 721–741, 1984.

[8] V. Upadhya and P. Sastry, "An overview of restricted Boltzmann machines," *Journal of the Indian Institute of Science*, pp. 1–12, 2019.

[9] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pp. 249–256, JMLR Workshop and Conference Proceedings, 2010.

[10] Y. Bengio, P. Lamblin, D. Popovici, H. Larochelle, *et al.*, "Greedy layer-wise training of deep networks," *Advances in Neural Information Processing Systems*, vol. 19, p. 153, 2007.

[11] M. G. Kendall, "A new measure of rank correlation," *Biometrika*, vol. 30, no. 1/2, pp. 81–93, 1938.

[12] X. Bi and H. Wang, "Early Alzheimer's disease diagnosis based on EEG spectral images using deep learning," *Neural Networks*, vol. 114, pp. 119–135, 2019.

[13] S. He, S. Wang, W. Lan, H. Fu, and Q. Ji, "Facial expression recognition using deep Boltzmann machine from thermal infrared images," in *2013 Humaine Association Conference on Affective Computing and Intelligent Interaction*, pp. 239–244, IEEE, 2013.

[14] R. Souriau, J. Lerbet, H. Chen, and V. Vigneron, "A review on generative Boltzmann networks applied to dynamic systems," *Mechanical Systems and Signal Processing*, vol. 147, p. 107072, 2021.

[15] G. W. Taylor and G. E. Hinton, "Factored conditional restricted Boltzmann machines for modeling motion style," in *Proceedings of the 26th annual International Conference on Machine Learning*, pp. 1025–1032, 2009.

[16] Y. Bengio, "Deep learning of representations: Looking forward," in *International Conference on Statistical Language and Speech Processing*, pp. 1–37, Springer, 2013.

Banca Conversacional Inteligente:

Democratizar los servicios financieros actuales y los nuevos (predictivos, prescriptivos y proactivos) con interacciones más fáciles y naturales que mejoren la Salud Financiera del cliente

Javier Porras Castaño
Doctorando Universidad Rey Juan Carlos, Madrid
Departamento: Ciencias de la Computación
Ronda, Málaga, España
@_javipc
www.linkedin.com/in/javipc
javiporrascastano@gmail.com

La Banca Conversacional Inteligente va a cambiar y revolucionar la forma que tienen los clientes de relacionarse con su banco y permitirán hacer realidad el reto de conseguir ofrecer la misma experiencia en cualquier canal y dispositivo (propios del banco o externos) gracias a la interacción mediante la interfaz más estándar, universal y sin curva de aprendizaje que es el lenguaje natural (texto o voz), con la propia forma de expresión de los clientes y con la que más cómodo se sienten. No sólo permitirá automatizar la atención al cliente sino también realizar cualquier operación de la banca digital actual y un conjunto de nuevos servicios financieros, que surgen gracias a la tecnología conversacional, que aportarán al cliente un valor añadido para mejorar su Salud Financiera. Mi tesis doctoral se centra en diseñar cuál es la arquitectura más adecuada para que los bancos puedan implantar esta Banca Conversacional Inteligente de forma ordenada, sostenible y escalable con los propios recursos de la organización, para ofrecer a los clientes servicios predictivos, prescriptivos y proactivos que le permitan mejorar su salud financiera anticipándole situaciones que le ayuden y aporten un valor añadido para mejorar la toma de decisiones y todo ello con interacciones naturales, cómodas y fáciles sea cual sea su perfil. Por último, veremos cómo se ha diseñado la arquitectura para permitir que el entrenamiento y reentrenamiento pueda ser realizado por perfiles no técnicos a través de grafos conversacionales visuales. Estos empleados de forma autónoma podrán crear y modificar los flujos conversacionales para que la escalabilidad de la funcionalidad sea ordenada y sostenible con los recursos del propio banco.

Innovación, Naturalidad, Omnicanalidad, Escalabilidad, Eficiencia, Grafos y Conversación

I. Introduction

Nos hemos parado a pensar qué tienen en común las aplicaciones más populares y con más volumen de usuarios digitales a nivel mundial como son el '*buscador de Google*', '*WhatsApp'*, '*Facebook'*, '*Siri'*, '*Alexa'* o *'Google Assistant'*. Efectivamente, su gran popularidad se debe a que permiten interactuar al usuario mediante lenguaje natural, con su propia forma de expresión, que es su forma inherente de comunicación y con la que más cómodo se siente [1].

Estas aplicaciones tan exitosas ofrecen la interfaz de interacción más sencilla y natural posible y sin curva de aprendizaje [2] para el usuario, sólo le requiere saber expresarse (ya sea con texto o con voz). Las pueden utilizar cualquier perfil de la sociedad, desde niños/as hasta la tercera edad, incluyendo a los que habitualmente tienen más dificultades al utilizar las tecnologías, cubrir las expectativas de los nativos digitales e incluir a los discapacitados. Las empresas, en plena transformación digital, están rediseñando sus Webs y Apps y están consiguiendo ofrecer una magnífica experiencia de usuario, sin embargo, no es lo que los usuarios quieren. No están teniendo en cuenta que los usuarios ya hemos demostrado que nos sentimos más cómodos interactuando en lenguaje natural que haciendo clic en una pantalla. WhatsApp y Facebook son las dos aplicaciones más utilizadas en el mundo y las empresas deberían tener en cuenta estos datos y seguir este modelo que ya está validado [3] y contrastado con los propios usuarios o clientes en vez de invertir recursos y esfuerzos en crear nueva una nueva web y app, bajo el modelo tradicional, haciendo clic en una pantalla y que desconoce si será bien aceptada por sus clientes. Por otro lado, el usuario ya no sólo utiliza su móvil para acceder a servicios digitales sino que está aceptando con naturalidad nuevos dispositivos como los '*smart watch'* o los '*altavoces inteligentes'* o incluso los más recientes que le permiten interactuar desde su coche, televisión o el espejo de su vestidor. Esta proliferación de nuevos dispositivos hace que sea inviable para las empresas tener que dedicar esfuerzos y recursos para ir integrándose en cada uno de estos nuevos dispositivos (y los que vayan apareciendo).

Los Sistemas Cognitivos Conversacionales son la solución perfecta para las empresas: permitirán atender a sus clientes con interacciones ágiles y naturales, a través de lenguaje natural y hacerlo en cualquier canal y dispositivo 24x7 ofreciendo una experiencia homogénea [4], gracias a permitir interactuar a los usuarios a través de interfaces conversacionales, ya sea con texto o voz: sea cual sea la funcionalidad a realizar, el momento, el canal y el dispositivo, el cliente siempre tendrá una experiencia homogénea con interacciones más ágiles, naturales y cómodas que le aporta un valor añadido para mejorar su Salud Financiera.

II. TESIS: DESCRIPCIÓN DE LA INVESTIGACIÓN

Gartner [5] predice que al finalizar 2021 el 85% de las interacciones con los clientes con sus empresas se atenderán de forma automática a través de Chatbots y Asistentes Virtuales (sin necesidad de que intervengan humanos). Juniper Research [6] pronostica que en 2023 la adopción de la tecnología conversacional en retail, banca o atención médica tendrá como resultado un ahorro de once millones de dólares. Estos datos son muy atractivos para el sector empresarial mundial y nos permite vislumbrar que, aunque ya hay muchas empresas que están incluyendo en sus estrategias los Chatbots y los Asistentes Virtuales, crecerá de forma exponencial en los próximos años.

El objetivo de mi tesis doctoral consiste en investigar y diseñar cómo las empresas deben adoptar los sistemas conversacionales cognitivos para generar eficiencia y ahorro en la cuenta de resultados de una forma sostenible y ordenada con sus recursos propios para que sea una inversión rentable y amortizable a medio plazo.

A. Definición: Sistema Conversacional Cognitivo

Se trata de una generalización de la Banca Conversacional Inteligente y es una pieza software que permite atender automáticamente a un cliente, a través de lenguaje natural (voz o texto), simulando una conversación humana sin necesidad de que tenga que intervenir ninguna persona de la empresa y no sólo para resolver consultas frecuentes (en el ámbito de la atención al cliente), sino para ofrecerle nuevos servicios de valor añadido inéditos hasta el momento (descriptivos, predictivos, prescriptivos y proactivos) justo cuándo y dónde lo necesita funcionando 24x7.

B. Contexto actual: brecha digital

En nuestro día a día los usuarios ya hemos adoptado con naturalidad la forma de interacción que nos ofrecen las aplicaciones digitales más populares para comunicarnos con la familia y amigos/as, principalmente WhatsApp y Facebook Messenger. A través de lenguaje natural, con nuestra forma de expresión utilizamos estas aplicaciones sin ningún tipo de problema y en muy poco tiempo ya somos usuarios expertos. Nos sentimos muy cómodos utilizándolas por la facilidad con la que nos permiten interactuar. Los usuarios ya demandamos poder hacerlo con la misma agilidad y naturalidad cuando interactuamos con cualquier empresa que nos provee un servicio, ya sea a nivel de atención al cliente o a nivel funcional para interactuar con nuestros productos y servicios. Existe una brecha digital [7] entre la forma en que interactúa el usuario con sus aplicaciones de ocio (interactúa a través de lenguaje natural) y cuando debe interactuar con una empresa (ésta le sigue ofreciendo canales tradicionales como el teléfono, correo, sms, web o app). Los clientes ya han demostrado que se sienten más cómodos interactuando en lenguaje natural que haciendo clic en una pantalla o tener que hacer una llamada a un call center. Las interfaces conversacionales ofrecen a las empresas la oportunidad de reducir esta brecha digital para permitir al cliente interactuar en lenguaje natural, tal y como éste utiliza el ecosistema de aplicaciones de su día a día (WhatsApp, Facebook Messenger, Siri, Alexa o Google Assistant).

C. Objetivos de la tesis

El principal objetivo de la tesis es diseñar cómo será la banca del futuro democratizando los servicios financieros para cualquier perfil con interacciones más naturales y ágiles para mejorar su Salud Financiera garantizando que la inversión para lograrlo sea sostenible y se pueda asumir con los recursos propios de las empresas. Concretamente, los tres objetivos son:

1. Definir la Arquitectura de la Banca Conversacional Inteligente
2. Definir la funcionalidad que ofrece la Banca Conversacional Inteligente. No sólo se ofrecerán los servicios de la banca digital actual sino servicios descriptivos, predictivos, prescriptivos y proactivos para mejorar la Salud Financiera del cliente.
3. Definir la Arquitectura de Grafos Conversacionales para que los perfiles no técnicos puedan crear y mantener los flujos conversacionales de la pieza software

D. Hipótesis de partida

El cliente aceptará con naturalidad interactuar con su Banco mediante lenguaje natural, ya sea voz o texto, tanto para resolver consultas (ámbito de atención al cliente) como para acceder y gestionar sus servicios y productos financieros personalizados.

De la misma forma que el cliente utiliza WhatsApp o Facebook Messenger para quedar con un familiar o amigo, o cuando utiliza Alexa, Siri o Google Assistant para tareas de su interés, podrá interactuar con su Banco mediante su forma de expresión de forma más fácil, sencilla y cómoda.

III. ORIGINALIDAD DE LA TESIS:

La originalidad de esta tesis radica en vislumbrar y concretar cómo será el futuro a medio plazo (6 años) de la Banca en España cambiando y revolucionando la relación entre los clientes y las entidades financieras [8]: se hará realidad el reto de la Banca personalizada y a medida para cada cliente permitiéndole interactuar con su propia forma de expresión que es con la que se siente más cómodo en cualquier canal y dispositivos.

La Banca Conversacional Inteligente supondrá un importante hito en la banca española (e internacional) porque pasaremos de tener interfaces digitales perfectamente definidas, a través de botones que realizan acciones concretas, a interfaces totalmente abiertas, con un campo de texto libre o un altavoz, donde el cliente puede solicitar cualquier aspecto, tarea, petición, etc y la Banca Conversacional debe estar preparada para atenderle y responderle adecuadamente. Algunos ejemplos de peticiones que puede solicitar el cliente son:

1. *'¿Cuánto pagaré de agua el próximo trimestre?'*
2. *'¿Cuánto gastaré en restaurantes en mis vacaciones?'*
3. *'¿Cuánto llevo gastado en gasolina este mes?'*
4. *'¿Me puedo permitir pagar un alquiler de 800€ al mes?`*
5. *'¿Cuándo me recomiendas que me compre el nuevo iphone?'*

Estas son algunas de las nuevas peticiones que podrá atender la Banca Conversacional Inteligente. Al ofrecer una interfaz totalmente abierta surgen nuevos servicios inherentes a esta forma disruptiva e innovadora de interactuar [9]: a través del lenguaje natural. En la actualidad, podemos consultar el saldo o hacer un bizum haciendo varios clic en los botones establecidos para ello en la app móvil, pero, por qué los clientes deben aprender el funcionamiento de esa app para gestionar sus productos financieros. Esto es lo que pretende solucionar la Banca Conversacional Inteligente y cada vez que reciba una petición, pregunta o solicitud del cliente, realizará siempre tres acciones y en este orden:

1: Comprender:

Cada una de las diferentes formas que tienen los clientes de solicitar una misma acción. Por ejemplo, algo tan simple como 'consultar el saldo' puede ser expresado de diferentes formas: 'dime cuánto dinero tengo', 'cuál es mi dinero' o 'cuál es mi capital'. El objetivo es entender qué necesita el cliente.

2: Resolver:

Una vez que el software ha comprendido la necesidad del cliente, estará capacitado para resolverlo. A través de invocaciones vía api REST con sistemas internos y externos, el software será capaz de resolver la acción solicitada por el cliente.

3: Responder:

Una vez que ya se ha realizado la solicitud del cliente, por último el software deberá generarle la respuesta en lenguaje natural (texto o voz). Aquí cada empresa definirá su estilo de atención al cliente, más o menos informal. En definitiva la forma de responder será un aspecto a definir a nivel corporativo ya que la Banca Conversacional Inteligente se convierte en un empleado más y debe responder de forma corporativa, según al estilo definido.

A continuación se detallan cómo lograr los tres objetivos propuestos en mi tesis doctoral:

A. Arquitectura Banca Conversacional Inteligente

Mi tesis define cuál es la mejor Arquitectura para que un Banco implante la Banca Conversacional Inteligente: no consiste en construir una pieza software que centralice todas las consultas/peticiones de los clientes porque sería muy difícil de mantener para los equipos de tecnología [10]. El planteamiento de la arquitectura está basado en la técnica divide y vencerás, contando con dos tipos de piezas software:

1. Muchos Chatbots (o voicebots). Cada uno de ellos será experto en un dominio de conocimiento muy concreto y acotado. Cuanto más pequeña sea la pieza software más fácil será de mantener.

2. Un solo Asistente Virtual que actúa como único canal de entrada hacia todos los chatbots (o voicebots) definidos. El cliente siempre interactúa con el Asistente Virtual y derivará, de forma transparente al cliente, la petición hacia el chatbot (o voicebot) que tiene la lógica necesaria para atenderle.

Esta Arquitectura de Banca Conversacional Inteligente esta basada para lograr la escalabilidad ordenada y sostenible en la empresa y tiene las siguientes características [11]:

1. Independiente de la tecnología que se utilice como motor NLP (propietaria o de un tercero)

2. Flexible porque en cualquier momento se puede cambiar de motor NLP [12] (de uno propio a una solución de un tercero o viceversa)

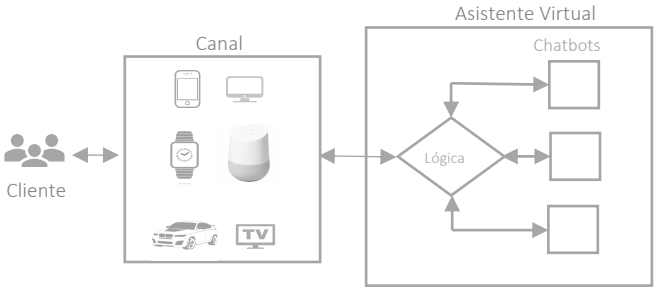
3. Escalable porque deforma fácil se pueden añadir nuevos chatbots (y voicebots)

4. Reutilizable porque es la misma para cliente interno y cliente final

5. Integrable en cualquier dispositivo y canal

6. Inclusiva porque permite interactuar con cualquier pieza CORE de la empresa

En la siguiente imagen se muestra un prototipo de esta Arquitectura de Asistente Virtual:



B. Servicios Predictivos, Prescriptivos y Proactivos

El objetivo es aportar al cliente diferentes visiones de su cuenta para anticipar situaciones que mejoren su Salud Financiera:

1. Visión Descriptiva:

   Es una visión del 'pasado': Analizar lo ocurrido en la cuenta del cliente para detectar patrones en su actividad financiera. Por ejemplo, éste podrá preguntar: *'¿Cuánto llevo gastado en gasolina este mes?'*

2. Visión Predictiva:

   Se trata de una visión del 'futuro': Anticipar situaciones que mejoren la toma de decisiones del cliente, gracias a analizar los datos del histórico de su actividad financiera [13]. Por ejemplo, éste podrá preguntar: *'¿Cuánto pagaré de agua el próximo trimestre?'*

   Principalmente se ofrecerá la posibilidad al cliente la posibilidad de predecir gastos de consumo de transacciones no periódicas como son los gastos del hogar (gas, luz o agua), gastos en la cesta de la compra y gastos derivados del ocio (restaurantes o espectáculos).

3. Visión Prescriptiva

   Se trata de una visión del 'presente': Una vez que tenemos la visión del pasado y del futuro podremos hacer recomendaciones personalizadas y a medida para el cliente que requiere en el momento actual [14]. Por ejemplo, éste podrá preguntar: '¿Me puedo permitir un alquiler de 800 € al mes?' Analizando el histórico de la actividad financiera del cliente y mezclando esta información con otros datos ajenos al cliente (abiertos o no), realizar recomendaciones a medida que resultan de su interés.

4. Visión Proactiva

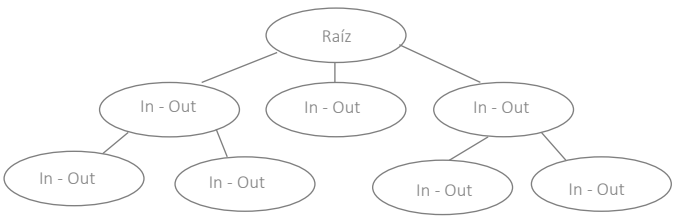
   Sin que el cliente lo solicite de forma explícita la Banca Conversacional Inteligente le aportará información de valor que le ayudará a tomar mejores decisiones y por lo tanto mejorar su Salud Financiera. Por ejemplo, si un cliente suele gastar todo el importe del crédito de su tarjeta antes del día 20 del mes, cuando el cliente haga una compra o consulte el saldo, el software le indicará si desea ampliar el límite de su tarjeta para poder llegar a final de mes tirando del crédito de su tarjeta.

Además, mezclando estas cuatro visiones la Banca Conversacional Inteligente ofrecerá un servicio innovador e inédito en el sector financiero hasta el momento: Comparar gastos con perfiles similares anonimizados. Habitualmente con familiares y amigos solemos comparar los gastos de consumo del hogar ('¿cuánto pagas tú de luz?'). El objetivo es que con este servicio se le ofrezca al cliente un indicador que le permita medir cómo son sus gastos y tratar de mejorar su comportamiento para mejorar su salud financiera.

C. Arquitectura de Grafos Conversacionales para diseñar los flujos conversacionales por perfiles no técnicos

En la empresa quien más conocimiento tiene de las tareas que se están realizando en cada sección o ámbito son aquellos/as quienes la hacen día a día, habitualmente perfiles no técnicos (no programadores). Dado que la Banca Conversacional Inteligente debe ofrecer al cliente una atención personalizada y a medida en diferentes aspectos: Tarjetas, Seguros, Hipotecas, Préstamos, etc, no hay nadie mejor para capacitar al software que cada uno de los perfiles especializados en estos ámbitos.

Por ello, se ha diseñado una Arquitectura de Grafos Conversacionales [15], que de forma visual y muy fácil e intuitiva permite a estos perfiles no técnicos realizar el entrenamiento y reentrenamiento del software para capacitarlo en aquellos ámbitos concretos en los que ellos son expertos. Por ejemplo, si el software debe atender al cliente sobre cualquier aspecto de Créditos al consumo, las personas que trabajan en ese departamento serán quienes diseñen los flujos conversacionales del software para que trate al cliente tal y como lo hicieran ellos mismos en persona; a continuación se muestra uno de ejemplo:



A través de esquemas visuales al anterior, los perfiles no técnicos podrán ir diseñando el flujo conversacional para cada conocimiento que se entrene en la Banca Conversacional Inteligente. De igual modo, esta arquitectura permitirá gestionar variables de contexto [16] que permita al software simular conversaciones lo más parecida a aquellas que tienen lugar entre humanos (simular como si el cliente estuviese hablando con su gestor). Las variables de contexto estarán disponibles para utilizarlas (crearlas, modificarlas o borrarlas) de forma muy gráfica y visual para facilitar la tarea. Se trata de un aspecto transformacional que las empresas tienen la oportunidad de aprovecharlo para incorporar al proceso de transformación digital a estos perfiles cuya experiencia es fundamental para realizar el entrenamiento del software.

En cada nodo del flujo conversacional se definen dos aspectos:

1. Intent: conjunto de diferentes formas de expresar una misma acción
2. Respuesta al Intent: la respuesta que dará la Banca Conversacional Inteligente cuando el cliente solicite alguna frase afín al intent definido.

Esta misma estructura se repetirá en todos los nodos del flujo conversacional. Por otro lado, tenemos las relaciones, que definen todos los posibles caminos que puede tener la conversación desde un nodo hacia el otro. Una conversación puede tener muchísimos caminos por lo que cuando más profundo (en vertical y horizontal) sea le grafo conversacional más humana será la conversación [17].

IV. METODOLOGÍA

Esta tesis doctoral se está realizando siguiendo una metodología 'agile' para lograr dos objetivos:

1: Realizar entregas rápidas y continuas: el proyecto de investigación se ha dividido en tareas más pequeñas y concretas planificadas cada dos semanas. Esta planificación ha sido consensuada con mi tutor de tesis y así el seguimiento por su parte es más ágil y efectivo.

2: Dotarlo de flexibilidad para adaptarlo a los aprendizajes e inferencias que voy descubriendo durante la investigación que puede afectar a la ruta definida en la planificación definida. Cualquier cambio en tareas planificadas inicialmente no tendrá impacto al trabajar con esta metodología.

Cada dos semanas se vuelven a planificar las nuevas tareas a realizar en el siguiente periodo, volviendo a incorporar aquellas que quedaran retrasadas o estuvieran bloqueadas por algún motivo. Tanto al tutor como a mi esta forma de trabajar nos está resultando bastante productiva, eficiente y fácil de seguir y medir para conocer el ritmo de progreso general del desarrollo de la tesis.

V. PLAN DE TRABAJO

El desarrollo de esta tesis doctoral comenzó en Octubre de 2019 y el plan de trabajo diseñado es el siguiente:

1. Diseñar la Arquitectura de un Sistema Conversacional Cognitivo (Banca Conversacional Inteligente) que permita al cliente acceder a servicios en tiempo real
2. Diseñar los servicios descriptivos, predictivos, prescriptivos y proactivos que ofrecerá la Banca Conversacional Inteligente al cliente.
3. Diseñar la Arquitectura de Grafos para dar soporte a los flujos conversacionales a través de una herramienta intuitiva que los puedan crear y mantener los perfiles no técnicos (no programadores)

VI. TAREAS

El plan de trabajo definido anteriormente se concreta en un conjunto de tareas específicas para cada uno de los tres objetivos propuestos:

1. *Arquitectura de un Sistema Conversacional Cognitivo (Banca Conversacional Inteligente):*

   Para acelerar los desarrollos estoy utilizando los servicios de Google Cloud [18] para un servidor en la nube (serverless) que se dimensiona automáticamente según las necesidades de atención de la arquitectura.

   Para la comprensión y generación del lenguaje natural estoy utilizando dos tipos de soluciones: desarrollo propio con Python (librería NTLK [19]) y soluciones comerciales (Dialogflow [20] y Rasa [21]). Estoy componiendo un informe comparativo de cada una de estas soluciones para tener una visión de 360º que permita a las empresas decidir qué opción es la más adecuada según sus necesidades.

2. *Servicios Descriptivos, Predictivos, Prescriptivos y Proactivos*

   Para desarrollar estos servicios nos vamos a basar en una base de datos y desarrollar un conjunto de modelos basados en machine learning que nos permitan ofrecer el valor añadido que mejore la Salud Financiera del cliente

   1. Base de datos: Se ha utilizado una base de datos de una entidad financiera de la República Checa que en 2012 publicó los datos movimientos de clientes anonimizados [22]. Contiene datos de 1.700 clientes entre los años 1992 y 1998. Contiene cinco tablas: Clientes, Cuentas, Tarjetas, Préstamos y Transacciones.
   2. Revisar y limpiar 'ruido': Realizar una revisión exhaustiva para eliminar aquellos datos que no aportan valor (por ejemplo, había algunas transacciones de cargo en cuenta de valor 0€ que han sido eliminadas o clientes que únicamente tenían informado su id pero ningún dato más).
   3. Análisis descriptivo de la base de datos: para descubrir correlaciones entre las variables, haciendo en primer lugar un análisis univariable para continuar con uno bivariable para deducir qué variables son las que necesitamos para nuestros servicios.
   4. Generar y validar modelos: Mediante desarrollos en Python utilizando librerías como Keras [23], TensorFlow [24] y Scikit Learn [25], sobre la plataforma de Google Cloud.
   5. Comparar y elegir el modelo más adecuado: que presente las mejores prestaciones para el modelo a desarrollar

3. Arquitectura de Grafos

   Para diseñar los flujos conversacionales esto investigando una solución basada en dos capas:

   1. Base de dato de Grafos: Probando para ello las siguientes tecnologías: Neo4j [26], GraphBd [27], Neptune [28] y Virtuoso [29]. Estoy realizando un informe comparativo de las ventajas y desventajas de cada una de ellas.
   2. Visualizador de grafos: Probando para ello las dos siguientes tecnologías: Linkurious [30] y TigerGraph [31]. Estoy realizando un informe comparativo de las ventajas y desventajas de cada una de ellas.

   Los perfiles de negocio siempre interactuarán con la herramienta visual y a medida que vayan 'pintando' los nodos y relaciones del grafo conversacional se actualizará de forma automática en la base de datos de grafo que subyace y que es totalmente transparente para ellos. La tecnología de grafos se adapta perfectamente a cada uno de los diferentes tipos de conversaciones que se podrían dar entre el cliente y la Banca Conversacional Cognitiva.

VII. CONCLUSIONES

Los Sistemas Conversacionales Cognitivos permitirán cambiar y revolucionar las relaciones de los clientes con las empresas permitiéndoles interactuar a través de como se sienten más cómodos y con su forma natural de comunicarse: el lenguaje natural. La Banca Conversacional Inteligente revolucionará el sector financiero permitiendo democratizar los servicios financieros a cualquier perfil de la sociedad eliminando la brecha digital y ofreciendo un nuevo conjunto de servicios que mejoran la Salud Financiera del Cliente en el momento y canal que ellos necesitan.

Hasta el momento era el cliente quien debía acudir al Banco (tenía que desplazarse hasta sus oficinas, navegar a su web o descargar su app para interactuar), sin embargo, la Banca Conversacional Inteligente permitirá un hito importante en la Banca, conseguir que estas relaciones sean bidireccionales para que el Banco se integre en el conjunto de aplicaciones que utiliza el cliente en su día a día: WhatsApp, Facebook Messenger, Siri, Alexa o Google Assistant.

Esta Banca Conversacional Inteligente podrá hacerse realidad con los propios recursos de las empresas porque el entrenamiento y mantenimiento del software será realizado por los perfiles no técnicos de las plantillas ya que ellos son los que tienen el conocimiento para capacitar al software. Para ello se habilitará una Arquitectura de Grafos Conversacionales que permitan diseñar los flujos conversacionales desde una interfaz visual y muy intuitiva.

AGRADECIMIENTOS

Quisiera agradecer a mi familia y amigos/as todo el apoyo que durante estos meses me están prestando para realizar esta investigación. Es mucho esfuerzo y tiempo el que estoy dedicando para lograr avanzar en los objetivos propuestos y todo ello es posible gracias a su empatía, cariño, apoyo y paciencia.

REFERENCES

[1] Krunal N. Patel. Robust and Secured Digital Audio Watermarking - Studies in Speech Signal Processing, Natural Language Understanding, and Machine Learning. Springer 2021, ISBN 978-3-030-53910-8, pp. 1-85.

[2] Aikaterini-Lida Kalouli. Hy-NLI: a Hybrid system for state-of-the-art Natural Language Inference. University of Konstanz, Germany, 2021.

[3] Adrien Bardet. Architectures neuronales multilingues pour le traitement automatique des langues naturelles. (Multilingual neural architectures for natural language processing). University of Maine, Le Mans, France, 2021.

[4] Chalermpol Tapsai, Herwig Unger, Phayung Meesad. Thai Natural Language Processing - Word Segmentation, Semantic Analysis, and Application. Studies in Computational Intelligence 918, Springer 2021, ISBN 978-3-030-56234-2, pp. 1-159.

[5] Garnert: Consultora de investigación de tecnologías: https://www.gartner.com/smarterwithgartner/top-priorities-for-finance-leaders-in-2021/

[6] Juniper Research: Revista de investigación: https://www.juniperresearch.com/press/chatbot-interactions-retail-reach-22-billion-2023

[7] Sanna Malinen, Aki Koivula, Ilkka Koiranen. How do Digital Divides Determine Social Media Users' Aspirations to Influence Others? SMSociety 2020: 141-147

[8] Giovanni Bonetta, Marco Roberti, Rossella Cancelliere, Patrick Gallinari. The Rare Word Issue in Natural Language Generation: A Character-Based Solution. Informatics 8(1): 20 (2021).

[9] Ting Hu, Christoph Meinel. Masked Hard Coverage Mechanism on Pointer-generator Network for Natural Language Generation. ICAART (2) 2021: 1177-1183.

[10] Sebastian Weigelt, Vanessa Steurer, Tobias Hey, Walter F. Tichy. Programming in Natural Language with fuSE: Synthesizing Methods from Spoken Utterances Using Deep Natural Language Understanding. Software Engineering 2021: 117-118.

[11] David Harbecke. Explaining Natural Language Processing Classifiers with Occlusion and Language Modeling. CoRR abs/2101.11889 (2021)

[12] Chidinma A. Nwafor, Ikechukwu E. Onyenwe. An Automated Multiple-Choice Question Generation Using Natural Language Processing Techniques. CoRR abs/2103.14757 (2021)

[13] Alena Moiseeva. Statistical natural language processing methods for intelligent process automation. Ludwig Maximilian University of Munich, Germany, 2020.

[14] Milan Gritta, Gerasimos Lampouras, Ignacio Iacobacci. Conversation Graph: Data Augmentation, Training and Evaluation for Non-Deterministic Dialogue Management. Trans. Assoc. Comput. Linguistics 9: 36-52 (2021).

[15] Yunlong Liang, Fandong Meng, Ying Zhang, Yufeng Chen, Jinan Xu, Jie Zhou. Infusing Multi-Source Knowledge with Heterogeneous Graph Neural Network for Emotional Conversation Generation. AAAI 2021: 13343-13352.

[16] Lan Zhang, Weihua Li, Quan Bai, Edmund Lai. Graph-based Self-Adaptive Conversational Agent. AAMAS 2021: 1791-1793.

[17] Joan Plepi, Endri Kacupaj, Kuldeep Singh, Harsh Thakkar, Jens Lehmann. Context Transformer with Stacked Pointer Networks for Conversational Question Answering over Knowledge Graphs. ESCW 2021: 356-371

[18] Plataforma Google Cloud: https://cloud.google.com

[19] Librería NLTK: software libre para procesamiento de lenguaje natural: https://www.nltk.org

[20] Dialogflow: Servicio de Google para procesamiento de lenguaje natural: https://cloud.google.com/dialogflow/docs

[21] Rasa: Software libre para procesamiento de lenguaje natural: https://rasa.com

[22] Base datos anonimizada Banco República Checa: https://datos.bancomundial.org/pais/republica-checa

[23] Keras: librería desarrollo modelos machine learning: https://keras.io

[24] TensorFlow: librería desarrollo modelos machine learning: https://www.tensorflow.org

[25] Scikit-learn: librería desarrollo modelos machine learning: https://scikit-learn.org/stable/

[26] Neo4j: Base de datos para grafos: https://neo4j.com/blog/announcing-graph-native-machine-learning-in-neo4j/

[27] GraphBd: Base de datos para grafos: https://graphdb.ontotext.com

[28] Neptune: Base de datos para grafos de Amazon: https://aws.amazon.com/es/neptune/

[29] Virtuoso: Base de datos para grafos: https://shop.openlinksw.com/license_generator/virtuoso-download/

[30] Linkurious: Herramienta para visualizar grafos: https://linkurio.us

[31] TigerGraph: Herramienta para visualizar grafos: https://www.tigergraph.com

Taxonomization of combinatorial optimization problems in Fourier space

Anne Elorza*
*Intelligent Systems Group
Department of Computer Science and Artificial Intelligence
University of the Basque Country UPV/EHU
San Sebastián, Spain
Email: anne.elorza@ehu.eus

Advisors: Leticia Hernando†, Jose A. Lozano*‡
†*Intelligent Systems Group*
Department of Mathematics
University of the Basque Country UPV/EHU
Leioa, Spain
‡*Basque Center for Applied Mathematics (BCAM)*
Bilbao, Spain
Emails: leticia.hernando@ehu.eus, ja.lozano@ehu.eus

Starting date: 1 March 2018

***Abstract*—In the field of combinatorial optimization, NP-hard problems represent a major challenge. This is because there is no known algorithm that solves this kind of problems for every possible instance in polynomial time on the dimension of the problem. To overcome this obstacle, a variety of metaheuristic algorithms have been proposed. One of the main difficulties when dealing with a specific problem is what algorithm to choose, since the performance of an algorithm can vary substantially from problem to problem, and even from instance to instance. So it would be desirable to find a matching between problem instances and algorithms, in such a way that the most suitable algorithm is assigned to solve a problem instance. A first step in this direction would be to create a taxonomy that groups problem instances that can be solved efficiently by the same algorithm. The aim of this thesis is to construct such a taxonomy in the case of permutation-based problems, by making use of the Fourier characteristics of each problem instance. As with the classical Fourier transform over the real line, which decomposes a function into a sum of sines and cosines, the Fourier transform over the symmetric group decomposes a permutation-based function into a linear combination of basis functions. Therefore, an objective function can be described through its Fourier coefficients, and we plan to use this information to create the taxonomy.**

***Index Terms*—combinatorial optimization, permutation-based, taxonomy, Fourier space, rankings**

This work is supported by the Basque Government (BERC 2018-2021, ELKARTEK program under the 3KIA project, KK-2020/00049, and IT1244-19) and from the Spanish Ministry of Economy and Competitiveness MINECO (projects PID2019-104966GB-I00/AEI/10.13039/501100011033 and PID2019-106453GA-I00/AEI/10.13039/501100011033). Jose A. Lozano acknowledges support by the Spanish Ministry of Science, Innovation and Universities through BCAM Severo Ochoa accreditation (SEV-2017-0718). Anne Elorza holds a predoctoral grant (ref. PIF17/293) from the University of the Basque Country.

I. Introduction

Combinatorial optimization refers to those optimization problems such that the search space is finite or countable infinite [14]. This kind of problems frequently arise in different fields of science and engineering. For instance, one could think of the problem of protein folding, the organization of tasks in a workshop or deciding the order in which airplanes should land in an airport. Among these problems, those that are named NP-hard should be highlighted. This is because there is no known algorithm that finds the best solution of all of the instances in polynomial time on the dimension of the problem. This means that the cost of exact methods becomes prohibitive as the dimension of the problem grows. As an alternative to exact methods, metaheuristic algorithms have largely succeeded in obtaining good quality solutions with reasonable computation times [1]. However, there remain certain issues that should be addressed in order to obtain an efficient application of these methods. Given the wide variety of algorithms that have been proposed, when one tries to solve a specific COP (combinatorial optimization problem), an initial and major difficulty that encounters is what algorithm to choose. The performance of an algorithm may vary substantially depending on the problem, and even, for the same problem, depending on the instance. An ideal scenario would be one in which, given a specific problem instance, one could select the most suitable algorithm to solve the given instance. A first step in this direction would be to create a taxonomy among COP instances, in such a way that instances that can be solved efficiently by the same algorithm are grouped together.

The aim of this thesis is to construct such a taxonomy in the case of a specific family of COPs: permutation-based COPs. These refer to those COPs whose solutions can be naturally codified as permutations, such as the linear ordering problem (LOP) [13], the traveling salesman problem (TSP) [11], the quadratic assignment problem (QAP) [12] or the permutation flowshop scheduling problem (PFSP) [7]. The first obstacle that one encounters when trying to taxonomize this kind of problems is their disparate definitions. Even though all of them are defined through matrices, the formulation of the problem and the meaning of the elements of the matrices vary from problem to problem. In one case, the elements of the entry matrix may represent distances between cities, in another case, they may represent the flows and distances between factories or they could concern tasks on machines. Even so, two instances of two different problems that appear to be completely different, could share certain properties and behave similarly under the same algorithm. What is more, they could even turn out to be identical in terms of the resolution of the problem (if the objective function value of all the solutions of the search space is the same for both instances).

From all the aforesaid, it becomes clear that, if the goal is to build a taxonomy that gathers instances of different problems, it could be useful to find a way of representing problems and problem instances in the same space, so that they can be easily compared with each other. For this purpose, we use the mathematical framework given by the Fourier transform (FT). Similarly to the classical FT, which converts real-valued functions into a sum of sines and cosines, the FT for the symmetric group decomposes functions defined over permutations into a sum of certain base functions [9]. In both cases, any function (in the case of COPs, this applies to fitness functions) can be univoquely determined by its Fourier coefficients. Although the FT is not a common tool in combinatorial optimization, it has been used to design a branch and bound algorithm in Fourier space [10]. Via the FT, the fitness functions of permutation-based COPs can be represented by a collection of Fourier coefficients. COPs whose fitness functions have different definitions could easily be put in the same terms through their Fourier coefficients, facilitating the comparison between problems and the creation of the taxonomy.

We would like to look for properties of the Fourier coefficients that influence the behaviour of optimization algorithms, and create the taxonomy on this basis. For example, it could happen that all of the instances that have a specific coefficient equal to zero behave similarly under a given algorithm, so all of the instances that have this coefficient equal to zero could be grouped together in the taxonomy. It should be noted that, if binary search spaces are considered, the Walsh transform (which is equivalent to the FT over permutations, but it applies to pseudo-Boolean functions) has already been used to carry out a classification of functions [4], [5]. However, the search spaces studied in that research have a reduced dimension ($n = 2, 3$).

Having established a mathematical tool that allows to compare different problems easily, there is a second obstacle to overcome: the large size of the space of permutation-based functions (which is infinite). To deal with this, we make a simplification that reduces substantially the number of functions. Note that most heuristic algorithms, such as local search or evolutionary algorithms that use tournament or ranking selection operators, do not make use of the exact value of the solutions in their machinery, but they only take into account the comparison between the objective value of the solutions (that is, they only check whether a solution is better or worse than another). If we limit our study to these algorithms, we can simplify the space of objective functions. For a given instance, instead of considering the values of the objective function, we view the function as a ranking of the elements of the search space, where the first element is the optimum, the second is the second best solution, and so on. There already exist studies that, instead of fitness functions, consider the rankings associated with them [8]. Note that, whereas the space of possible permutation-based functions is infinite, the space of rankings is finite. If the solutions are permutations of size n, and we only consider injective functions, there are $n!$ different permutations and $(n!)!$ possible rankings. We hypothesize that problems that share a high number of rankings should behave similarly under the same algorithms. This is why one of the objectives of this PhD is to study the intersections of rankings that belong to different COPs, and take this information into account in the creation of the taxonomy. Let us recall that we restrict our study to the aforementioned algorithms: the ones that, in their machinery, only take into account comparisons of objective function values.

The heart of this PhD is represented in Figure 1. The idea is to explore the relationships between the main three concepts that are shown in the figure. Looking at COPs as rankings or through their Fourier coefficients has different benefits. The rankings can be related to algorithms more easily, since the performance of the algorithms that we are considering depends only on the ranking. On the other hand, Fourier coefficients allow to express any COP in the same terms and it becomes easier to compare problems with different representations. Although the taxonomy of problem instances would be defined by giving conditions in Fourier space, we intend to create it by taking into account the relationship between rankings and Fourier coefficients. The final goal of this PhD would be the creation of the taxonomy, based on the analysis of the relationship between the concepts represented in Figure 1, which is in itself almost a completely unexplored area.

II. Objectives

The main goal of this thesis is to analyze the relationship between the concepts represented in Figure 1, with a view to create a taxonomy among instances of permutation-based COPs, such that instances that can be solved efficiently by the same algorithms are grouped together. The initial objectives that we established in order to achieve this goal are the following:

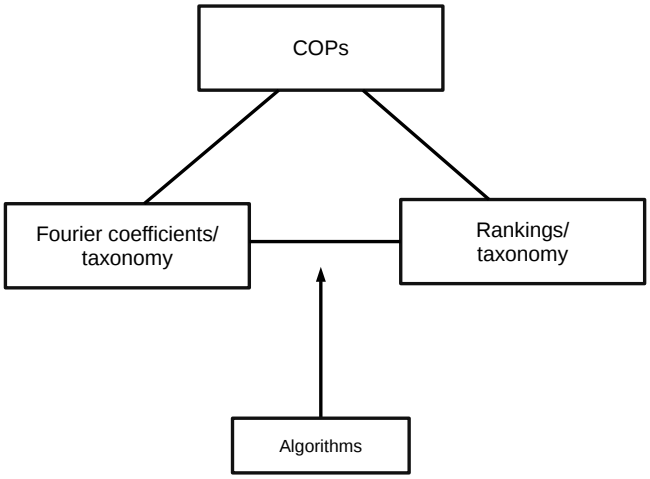


Fig. 1: This thesis explores the relationship between the three concepts that are framed by squares: COPs, Fourier coefficients and rankings. The goal is to create a taxonomy that helps to achieve a more efficient application of algorithms.

O1 Analysis and characterization of COPs in Fourier space. This objective aims at tracing the shape of the Fourier coefficients associated with different COPs. A previous study has shown that the objective function of a QAP instance has only 4 non-zero Fourier coefficients [10]. It would be desirable to have a more exact description of the Fourier coefficients of the objective functions of different COPs. The COPs considered are the LOP, TSP, QAP and PFSP.

O2 Analysis and characterization of rankings associated with COPs and Fourier coefficients. This objective has two main parts. On the one hand, the rankings that are associated with different COPs should be characterized. On the other hand, the effect of the different Fourier coefficients on the generation of rankings should be analyzed.

O3 Analysis and characterization of the intersection of the rankings generated by different COPs. With a view to the creation of the taxonomy, the idea is that problems that share many rankings should behave similarly under the same algorithm. Therefore, the intersections between the sets of rankings generated by different COPs should be studied. In addition, we would like to study the relationship between this intersections and their Fourier coefficients.

O4 Creation of the taxonomy and evaluation of the goodness of different algorithms in the groups of the taxonomy. The taxonomy would be defined in Fourier space by taking into account the information obtained in the previous objectives. To illustrate how the taxonomy could be defined with a hypothetical example, instances with a certain coefficient equal to zero could be grouped together. The behaviour of different algorithms over each of the sets of the taxonomy should be evaluated.

III. Methodology and work plan

As the research has progressed, the findings that we have made have arised new relevant research questions, which we had not initially considered. This has led us to carrying out more tasks than what we originally planned. At present, we have completed objective **O1**, and we have also made considerable progress in the fulfilment of objective **O2**. We plan to write a paper related to each of the four objectives and send them to a referred journal of the JCR. Preliminary results may also be presented in conferences. The tasks related to each of the objectives are detailed below:

Completed objective:

O1 Analysis and characterization of COPs in Fourier space.

- T1 Literature review focused on the FT over permutations and its applications to combinatorial optimization and related fields.
- T2 Taking the study of [10] as a basis, determine which Fourier coefficients are zero in other permutation-based COPs such as the LOP, the TSP (both the symmetric and the non-symmetric version) and the PFSP.
- T3 Find an exact characterization of the LOP, TSP (both the symmetric and the non-symmetric version) and PFSP in Fourier space.
- T4 Study the intrinsic dimension of different COPs, which represents the minimum number of parameters needed to define an instance of a problem.
- T5 Study the intersection of COPs. If a COP is understood as a set of functions (the objective functions associated to all of the possible instances), we can view the intersection between two COPs as the subset composed of the functions that belong to both sets.
- T6 Write a paper that contains the characterization of the COPs that have been studied.

NOTE: A paper with the results obtained in the fulfilment of this objective has been sent to *Evolutionary Computation Journal*. Tasks T4 and T5 were not originally planned, but followed naturally after the findings made with the completion of task T3.

Objectives to develop/in progress:

O2 Analysis and characterization of rankings associated with COPs and Fourier coefficients.

- T1 Study the rankings that are generated by the LOP, TSP, QAP and PFSP. Find a description as accurate and simple as possible of the rankings generated by each of these problems. In addition, the number of rankings that are generated by each problem should be determined or bounded.
- T2 Study what rankings are generated by different COPs when restrictions are imposed on the Fourier coeffi-

cients of the objective functions (for example, setting one coefficient to 0).

T3 Write a paper that summarizes the findings of the last two tasks.

NOTE: The fulfilment of this objective is at an advanced stage and, at present, we are writing a paper that summarizes the results obtained in the process.

O3 Analysis and characterization of the intersection of the rankings generated by different COPs.

T1 Study the intersections of the rankings that are generated by the LOP, TSP, QAP and PFSP. Find a description as accurate and simple as possible of the rankings that belong to the intersection of the rankings of two or more of these problems. In addition, the number of rankings that belong to these intersections should be determined or bounded.

T2 Write a paper that describes the intersections of rankings associated with different COPs.

O4 Creation of the taxonomy and evaluation of the goodness of different algorithms in the groups of the taxonomy.

T1 Define a taxonomy where each set is created taking into account the intersections found in the achievement of objective O3.

T2 Analyze the behaviour of local search algorithms over each of the sets of the taxonomy under different neighborhoods. Common neighborhoods that appear in the literature, such as 2-exchange or insert, will be considered. Other algorithms, such as estimation of distribution algorithms or genetic algorithms, could be included.

T3 Write a paper that describes the taxonomy and shows the behaviour of different algorithms over the sets of this taxonomy.

IV. Relevance

Even though metaheuristic algorithms have largely succeeded in the resolution of COPs, there still exist a number of aspects that require a deeper study in order to obtain a more efficient application. In this sense, one of the major problems is how to select the most appropriate algorithm. Given a problem or, even more specifically, a problem instance, it would be helpful to be able to detect in advance which features should an algorithm have to find a good solution to the problem. Under these circumstances, one would be able to select the algorithm. Finding such a matching between problem instances and algorithms is by no means simple, but even a small step in this direction could be of great help. For instance, knowing which neighborhood is the most suitable to solve a COP instance through a local search would imply significant time savings. This is why we believe that the creation of a taxonomy that groups problem instances, together with the preliminary results that we may obtain, could have a considerable practical impact.

What has been mentioned applies to the final goal of this PhD, which is the creation of the taxonomy, but there remain other aspects worth to be commented. To start with, there only exist a few studies that relate COPs to the FT [10], [15], which means that we open a research line that has been mainly unexplored. Similarly to elementary landscape decomposition, the FT decomposes the fitness function into a sum of orthogonal components. We have observed that the number of non-zero Fourier coefficients of the QAP, the LOP and the symmetric and asymmetric TSP (if the coefficient representing the mean value of the function is discarded) coincides with the number of components of the elementary landscape decompositions found in the literature [2], [3], [6], [16]. This evidence suggests that the Fourier decomposition could be closely related to elementary landscape decomposition. We conjecture that it could be possible to find a neighborhood such that each of the Fourier components is an elementary landscape with respect to the given neighborhood. Due to this relationship between Fourier and elementary landscape decomposition, which is a well-established research area, we believe that the study of COPs in Fourier space is a promising research line.

References

[1] Mohamed Abdel-Basset, Laila Abdel-Fatah, and Arun Kumar Sangaiah. Metaheuristic algorithms: A comprehensive review. *Computational intelligence for multimedia big data on the cloud with engineering applications*, pages 185–231, 2018.

[2] Josu Ceberio, Borja Calvo, Alexander Mendiburu, and Jose A Lozano. Multi-objectivising combinatorial optimisation problems by means of elementary landscape decompositions. *Evolutionary computation*, 27(2):291–311, 2019.

[3] Francisco Chicano, L Darrell Whitley, and Enrique Alba. A methodology to find the elementary landscape decomposition of combinatorial optimization problems. *Evolutionary Computation*, 19(4):597–637, 2011.

[4] Lee A Christie. The role of walsh structure and ordinal linkage in the optimisation of pseudo-boolean functions under monotonicity invariance. 2016.

[5] Lee A Christie, John AW McCall, and David P Lonie. Minimal walsh structure and ordinal linkage of monotonicity-invariant function classes on bit strings. In *Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation*, pages 333–340, 2014.

[6] Anne Elorza, Leticia Hernando, and Jose A Lozano. Taxonomization of combinatorial optimization problems in fourier space. *arXiv preprint arXiv:1905.10852*, 2019.

[7] Jose M Framinan, Jatinder ND Gupta, and Rainer Leisten. A review and classification of heuristics for permutation flow-shop scheduling with makespan objective. *Journal of the Operational Research Society*, 55(12):1243–1255, 2004.

[8] Leticia Hernando, Alexander Mendiburu, and Jose A Lozano. Characterising the rankings produced by combinatorial optimisation problems and finding their intersections. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 266–273, 2019.

[9] Jonathan Huang, Carlos Guestrin, and Leonidas Guibas. Fourier theoretic probabilistic inference over permutations. *Journal of machine learning research*, 10(May):997–1070, 2009.

[10] Risi Kondor. A fourier space algorithm for solving quadratic assignment problems. In *Proceedings of the twenty-first annual ACM-SIAM symposium on Discrete Algorithms*, pages 1017–1028. SIAM, 2010.

[11] Gilbert Laporte. The traveling salesman problem: An overview of exact and approximate algorithms. *European Journal of Operational Research*, 59(2):231–247, 1992.

[12] Eliane Maria Loiola, Nair Maria Maia de Abreu, Paulo Oswaldo Boaventura-Netto, Peter Hahn, and Tania Querido. A survey for the quadratic assignment problem. *European journal of operational research*, 176(2):657–690, 2007.

[13] Rafael Martí and Gerhard Reinelt. *The linear ordering problem: exact and heuristic methods in combinatorial optimization*, volume 175. Springer Science & Business Media, 2011.

[14] Christos H Papadimitriou and Kenneth Steiglitz. *Combinatorial optimization: algorithms and complexity*. Courier Corporation, 1998.
[15] Dan Rockmore, Peter Kostelec, Wim Hordijk, and Peter F Stadler. Fast fourier transform for fitness landscapes. *Applied and Computational Harmonic Analysis*, 12(1):57–76, 2002.
[16] Peter F Stadler. Landscapes and their correlation functions. *Journal of Mathematical chemistry*, 20(1):1–45, 1996.

Minería de datos para la sostenibilidad urbana

14 de junio de 2021

Francisco Rodríguez-Gómez
Leng. y Ciencias de la Computación
Universidad de Málaga
Málaga, España
francisco.rdg.gmz@uma.es

Dirigido por: Llanos Mora-López
Leng. y Ciencias de la Computación
Universidad de Málaga
Málaga, España
llanos@uma.es

Dirigido por: José del Campo-Ávila
Leng. y Ciencias de la Computación
Universidad de Málaga
Málaga, España
jcampo@uma.es

***Resumen*—Este proyecto de tesis pretende avanzar en la lucha contra el cambio climático desde la mejora de la sostenibilidad urbana gracias a la aplicación de procesos de minería de datos. En ese proceso se combinarán técnicas conocidas para dar lugar a metodologías novedosas, al mismo tiempo que se propondrán nuevos métodos en al ámbito del aprendizaje automático. En ese proceso se incorporará el conocimiento de los expertos, de forma que los resultados de esa minería de datos puedan ser realmente útiles. Las aportaciones desarrolladas se canalizarán en forma de herramientas software gratuitas.**

***Términos de búsqueda*—Aprendizaje computacional, cambio de concepto, energía fotovoltaica, naturación urbana, islas de calor urbano, patrones de consumo eléctrico**

I. Resumen

En la siguiente sección se realiza una descripción del problema a tratar, de los objetivos generales y específicos, así como de la investigación a realizar, haciendo una revisión del estado del arte en cuanto a las investigaciones previas que la sustentan, así como las aportaciones que pretenden hacerse con el proyecto de tesis.

I-A. Introducción

Debido al calentamiento global producido por la emisión de gases de efecto invernadero en las ciudades, los seres vivos y el planeta están sufriendo sus terribles consecuencias: enfermedades, sequías, derretimiento de los glaciares, extinción de especies, altas temperaturas... La sostenibilidad urbana consiste en el compromiso de las personas en cuanto a la utilización de medios urbanos que no degraden el entorno y que disminuyan su impacto en el mismo, mejorando así la calidad de vida de los seres vivos sin poner en compromiso a las futuras generaciones. Para mejorar la sostenibilidad urbana e intentar frenar el avance del cambio climático es de vital importancia el desarrollo de herramientas que ayuden en su lucha.

Por una parte, sería interesante disponer de herramientas capaces de detectar automáticamente las zonas más castigadas por el calentamiento global en las diferentes ciudades, que permitieran determinar el nivel de deterioro, determinando así las zonas prioritarias sobre las que actuar. Una de las alternativas para mitigar las altas temperaturas de dichas zonas más desfavorables sería la naturación de la ciudad añadiendo infraestructuras verdes (techos verdes, fachadas verdes, parques, jardines ...), por lo que también sería ideal disponer de herramientas que pudieran ayudar en la toma de decisiones en cuanto a cuales podrían ser las ubicaciones óptimas donde realizar la naturación y cual sería la tipología más adecuada.

Por otro lado, sería muy favorable sustituir el uso de energías contaminantes que producen gases de efecto invernadero, por energías renovables como la energía solar, por lo que también podría ser importante disponer de herramientas que de forma automática pudieran asesorar en cuanto a las posibles áreas óptimas en las que colocar instalaciones fotovoltaicas en las ciudades, lo cual podría favorecer la sustitución del uso de energías contaminantes por este tipo de energías renovables.

También sería de gran utilidad encontrar una metodología y los modelos adecuados para determinar los patrones de consumo eléctrico en los que clasificar a los consumidores y modelos que permitan clasificarlos, lo cual derivaría en una mejor gestión energética y por lo tanto repercutiría en la mejora de la sostenibilidad urbana.

I-B. Objetivos

I-B1. Generales:

1. Desarrollar nuevas metodologías para generar nuevo conocimiento e integrarlo en herramientas basadas en modelos de aprendizaje automático que puedan servir de ayuda en la lucha contra el cambio climático y en la mejora de la sostenibilidad urbana.

I-B2. Específicos:

1. *Modelizado_zonas_UHI.* Desarrollar modelos basados en inteligencia artificial y el conocimiento de los expertos que de forma automática permita al usuario conocer las áreas de interés urbano más desfavorecidas por el UHI[1] y el nivel de deterioro de las mismas, analizando la relación entre la temperatura y vegetación del entorno. Este objetivo será el punto de partida del siguiente.
2. *Modelizado_naturacion.* Desarrollar modelos basados en la inteligencia artificial, que permitan de forma automática al usuario conocer las posibles zonas óptimas en

[1]Una isla de calor urbana (UHI) es un área urbana o área metropolitana que es significativamente más cálida que las áreas rurales circundantes debido a las actividades humanas

las que implantar las formas de naturación urbana más idóneas (techos verdes, fachadas verdes, ajardinar suelo...) decidiendo la tipología de especies vegetales más adecuadas (árboles, arbustos, césped...) para la mitigación del UHI en las zonas más desfavorecidas.

3. *Modelizado_predicciones_energia_solar.* Desarrollar modelos que de forma automática permitan al usuario conocer la producción fotovoltaica a corto plazo (producción para el día siguiente) y a largo plazo (producción media diaria) esperada en un área de interés urbana: barrios, calles... facilitando así el proceso de toma de decisiones en cuanto a la posible ubicación óptima para realizar las instalaciones.
4. *Modelizado_patrones_consumo_electrico.* Desarrollar una nueva metodología basada en modelos de aprendizaje automático que permita determinar el número óptimo de patrones de consumo horario que modelen el consumo eléctrico, así como determinar dichos patrones de consumo, y que permita la clasificación de los consumidores según dichos patrones. Este conocimiento sería importante para los consumidores y para las propias compañías de suministro eléctrico, en cuanto al ajuste y personalización de tarifas y por la posible mejora en la planificación de la energía que se debe producir para satisfacer las necesidades de consumo.
5. *Cambio de concepto.* Análisis de la viabilidad de utilizar modelos de detección de cambio de concepto para caracterizar los cambios de comportamiento en el ámbito de la sostenibilidad urbana y la lucha contra el cambio climático.
6. *Implementación.* Implementación de los modelos y metodologías anteriormente descritas que permitan satisfacer los objetivos de la investigación.

I-C. Estado del arte

A continuación, se describe el estado del arte en cuanto a sostenibilidad urbana y a minería de datos para el desarrollo de modelos y herramientas inteligentes para la sostenibilidad urbana.

I-C1. Sostenibilidad urbana: Como se ha descrito en la sección I-A, existen múltiples factores que pueden influir en la mejora de la sostenibilidad urbana y ayudar en la lucha contra el cambio climático. Los objetivos de la tesis propuesta se centran en la generación de conocimiento, metodologías, y herramientas que ayuden a mitigar el efecto de las islas de calor urbano y favorezcan la sustitución de energías contaminantes por energías renovables como la energía solar.

Se pueden encontrar múltiples estudios que predicen un incremento generalizado de las temperaturas debido al calentamiento global, que sería más intenso en zonas urbanas, debido al conocido efecto isla de calor urbano. Sus efectos tienen impacto en la salud de los seres vivos, en su calidad de vida, e incluso en las infraestructuras instaladas para su mitigación [1], [2]. Se deben buscar alternativas para paliar los efectos del cambio climático [3] mientras no puedan reducirse o eliminarse el uso de energías contaminantes productoras de gases de efecto invernadero. Para la mitigación del UHI, está teniendo un creciente interés la instalación de infraestructuras verdes [4], entre las que se pueden destacar los techos y fachadas verdes [5].

La disponibilidad de zonas para la naturación urbana es limitada. Se estima que en zonas densamente pobladas las cubiertas de los edificios suponen entre el 20 % y el 25 % de la superficie total [6], por lo que sería deseable disponer de herramientas que pudiesen sugerir las ubicaciones más adecuadas para su instalación para la mitigación del UHI en las zonas más desfavorables. Es necesario trabajar en términos de localización y diseño porque se ha comprobado su relación con los efectos observados [7]. Los casos de estudio que se han presentado por los expertos en infraestructuras verdes son aplicados a ciudades concretas, y siguiendo un proceso manual o poco automatizado [8], [9], por lo que la minería de datos sería una herramienta útil para la automatización del proceso de extracción de conocimiento.

La reducción de la emisión de los gases de efecto invernadero emitidos a la atmósfera es una acción clave para la lucha contra el cambio climático y para la mejora de la sostenibilidad urbana. La implantación de energías renovables está consiguiendo desplazar a las energías tradicionales (basadas en combustibles fósiles o nucleares) y son un recurso imprescindible para la mejora. Por las características comunes que tienen la mayor parte de las ciudades, la energía renovable que mejor permite una generación localizada en la propia ciudad es la energía solar. Estas instalaciones distribuidas en las ciudades (ya sea por particulares, empresas o instituciones) buscan el autoabastecimiento. El interés de la sociedad en este tipo de instalaciones va incrementando debido a sus beneficios económicos, pero también por una mayor concienciación que hay por utilizar energías renovables. Si sigue percibiéndose la mejora económica y ambiental que suponen dichas instalaciones, es de esperar, que aumente su demanda. La predicción de la producción que una instalación fotovoltaica generará a corto y a largo plazo permitirá trasladar esos beneficios. Existen alternativas en esta línea como pueden ser Google Sun Roof (gratuita en EE.UU.), y HuellaSolar (gratuita en algunas ciudades de España). Podría ampliarse el abanico de aplicaciones disponibles con una nueva herramienta que de forma gratuita permita estimar la producción fotovoltaica a corto y largo plazo en área urbanas: barrios, calles...

La estimación a corto plazo aportará a los consumidores particulares información que les permitirá ajustar sus comportamientos de consumo diariamente. Para hacer esas predicciones diarias de producción eléctrica es necesario disponer de predicciones de la radiación solar. Desde el campo de la minería de datos también se puede contribuir a esta tarea de la predicción de la radiación solar [10], [11]. Recientemente se han propuesto distintos patrones de radiación diaria, lo que permite ajustar mejor los modelos y conseguir predicciones más precisas [12].

I-C2. Minería de datos: El desarrollo de modelos que sean capaces de obtener el conocimiento latente en los datos es el punto de partida de la creación de sistemas inteligentes. La

dificultad para la elaboración y la validación de los mismos radica en el volumen de datos y el número de variables que intervienen. Como se describe en II-A, la minería de datos es una metodología que facilita este proceso, ya que entre sus fases encontramos una fase de filtrado de los datos necesarios (fase de preparación de datos), y una fase de generación de modelos pasa obtener el conocimiento oculto en los datos.

Como ejemplos de aplicación de la minería de datos a la sostenibilidad urbana, se encuentran varios ejemplos: modelos predictivos basados en los datos para estimar el uso de energía en los edificios a escala urbana [13], integración de diferentes fuentes de datos para extraer nuevo conocimiento [14] o cálculo de las mejores localizaciones para nuevas instalaciones [15].

Los métodos que se utilizan en la fase de modelizado del proceso de minería de datos son un elemento fundamental para la obtención de resultados. En el contexto de la sostenibilidad urbana, se utilizan las mismas técnicas que se puedan utilizar en cualquier otro dominio: aprendizaje supervisado, no supervisado o una combinación de ambos [16], [17], empleando algoritmos para estimar diferentes modelos (clustering, regresiones lineales, redes neuronales, sistemas multiclasificadores, sistemas híbridos, etc.) [18]–[20], integrando datos de diferentes fuentes (imágenes, datos meteorológicos, sensores, etc.) [14], [20]. Al ser los entornos reales cambiantes, como es el caso de la sostenibilidad urbana [16], [21], [22], los datos no son estacionarios y, por lo tanto, los modelos deberían de ser capaces de detectar y adaptarse a dichos cambios para que sigan teniendo validez. En el campo de la minería de datos, existen métodos para detectar los cambios de concepto [23] y para adaptar el aprendizaje de forma que siempre se disponga de un modelo actualizado [24].

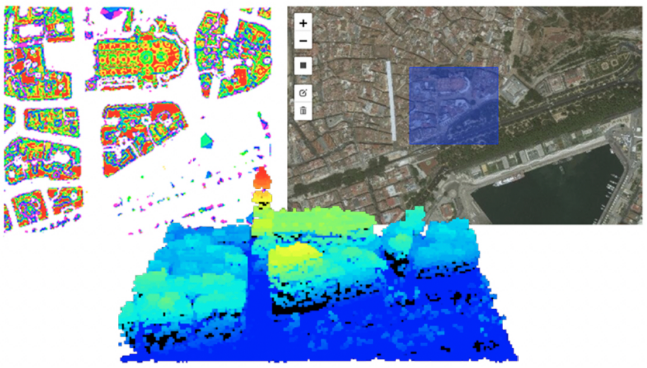


Figura 1. Estimación de energía fotovoltaica en un área de interés de Málaga. Fuente: elaboración propia

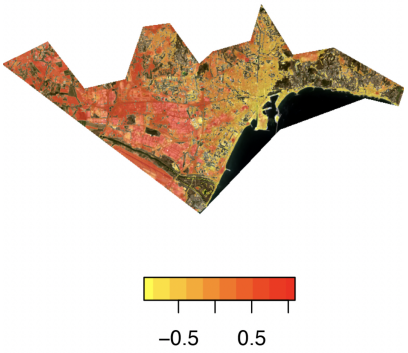


Figura 2. Zonas castigadas por el UHI y nivel de deterioro en un área de interés de Málaga. Fuente: elaboración propia

I-D. Contribución

La principal contribución de la tesis sería el desarrollo y difusión de herramientas gratuitas que pudieran ser de ayuda en la lucha contra el cambio climático y en la mejora de la sostenibilidad urbana, la generación de conocimiento para los expertos, y la creación de nuevos algoritmos o modificación de algunos existentes, que permita la solución de problemas específicos del dominio. (cambio de concepto, clustering...).

Las herramientas que se pretenden desarrollar ofrecerán las siguientes funcionalidades:

1. Estimación de la energía solar fotovoltaica media diaria y para el día siguiente en áreas de interés urbano.
2. Posibles ubicaciones óptimas de instalaciones fotovoltaicas en áreas de interés urbano: barrios, calles... La Figura 1 muestra captura de la información que manejaría una posible herramienta.
3. Detección automática de las zonas urbanas más desfavorables por el calentamiento global (UHI).
4. Información del nivel de deterioro de las zonas más castigadas por el UHI. La Figura 2 muestra una captura de la herramienta que muestre el nivel de deterioro de las zonas más desfavorecidas.
5. Posibles ubicaciones óptimas para la naturación urbana (techos verdes, suelos...) para mitigar las zonas más castigadas por el calentamiento global e información sobre el tipo de vegetación más adecuada. La Figura 3 muestra diferentes formas de naturación urbana: techos verdes, fachadas verdes, parques, jardines...
6. Determinación del número óptimo de clusters (patrones de consumo horario) que modelen el consumo de energía eléctrica, así como de los patrones en sí.
7. Clasificación de los consumidores según los patrones propuestos en el punto anterior.

Dichas herramientas serían potencialmente útiles para las administraciones públicas, planificadores urbanos (*urban planners*), ayuntamientos..., ya que les podrían ayudar en la toma de decisiones relativas a acciones encaminadas a la mejora de la sostenibilidad urbana y a la lucha contra el cambio climático. Por ejemplo, los urban planners podrían, con sólo seleccionar un área de interés en una ciudad, conocer sobre que áreas es más urgente actuar, y cuales serían las zonas óptimas dónde ubicar vegetación (suelos, tejados verdes...) para frenar el avance del cambio climático. Los ayuntamientos podrían saber seleccionando diferentes áreas de interés urbana, dónde les convendría más realizar instalaciones de sistemas



Figura 3. Formas de enverdecimiento de la ciudad. Fuente: https://www.cnbc.com/green-cities-rewilding-techniques-in-urban-areas/

fotovoltaicos, y conocer a priori, si cubrirían la demanda de energía necesaria. Esta herramienta podría ayudar a que se sustituyan a mayor escala las energías contaminantes por energías renovables, consiguiendo ciudades más sostenibles y menos castigadas por el cambio climático.

Para la implementación de las herramientas anteriormente descritas, se plantea proponer nuevos algoritmos o modificaciones de alguno existente para solucionar problemas específicos del dominio (clustering, cambio de concepto...)

II. Metodología y plan de trabajo

En la siguiente sección, se realiza un breve resumen de la metodología seleccionada para cumplir los objetivos del proyecto de tesis, así como una breve descripción de la planificación temporal asignada a las tareas.

II-A. Metodología

Existen diversas metodologías para llevar a cabo un proceso de minería de datos (MDP [25], TDSP [26] o CRISP-DM [27]). Es importante seguir una metodología que ofrezca un marco de trabajo conocido y seguro que sirva de guía a lo largo del proceso. La metodología que se va a seguir para la consecución de los objetivos, va a ser CRISP-DM, por ser la más utilizada en proyectos de minería de datos y unificar las ventajas de la otras metodologías.

La Figura 4 muestra a modo de esquema, las principales fases y tareas de la metodología. Las fases son las siguientes:

Fase 1 *Comprensión del problema.* Se estudia el problema a tratar, y una vez comprendido, se fijan los objetivos del proceso de minería de datos, y se establece un plan.

Fase 2 *Comprensión de los datos.* Estudiar los datos necesarios así como decidir sus fuentes, sin perder de vista los objetivos del proceso de minería de datos.

Fase 3 *Preparación de los datos.* Preparar los conjuntos de datos necesarios a partir de los datos de las fuentes seleccionadas en el apartado anterior.

Fase 4 *Modelizado.* Utilización de técnicas de aprendizaje automático para obtener conocimiento a partir de los datos preparados en el apartado anterior.

Fase 5 *Evaluación e interpretación.* Expertos en el dominio del problema evalúan los resultados obtenidos en cuanto a la satisfacción de los objetivos inicialmente definidos.

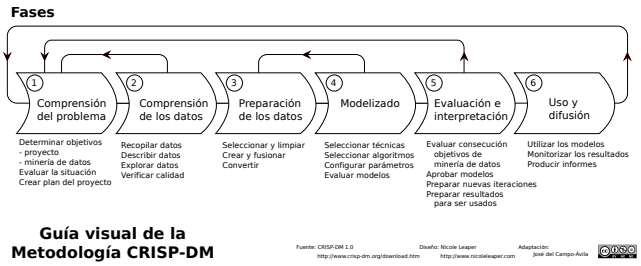


Figura 4. Metodología CRISP-DM. Fases y tareas generales.

Fase 6 *Difusión y uso.* El nuevo conocimiento será integrado en forma de herramienta software conducida por datos y se difundirá mediante publicaciones en revistas científicas de alto impacto.

La metodología CRISP-DM sigue un proceso iterativo en el que pueden darse varias vueltas para mejorar las iteraciones anteriores.

II-B. Plan de trabajo

La estimación temporal será de 36 meses aproximadamente.

El plan de trabajo estará enfocado a desarrollar las metodologías, conocimiento, modelos, y herramientas basados en la minería de datos para satisfacer los objetivos descritos en I-B siguiendo la metodología CRISP-DM descrita previamente en la sección II-A. Finalmente, se difundirán los resultados y se realizará la transferencia de conocimiento, en forma de publicaciones en revistas de alto impacto y en forma de herramientas software.

Para alcanzar los objetivos propuestos, se seguirán los siguientes pasos:

1. Se estudiará el problema.
2. Se estudiarán las posibles fuentes de datos y los propios datos.
3. Se realizará la preparación y transformación de los mismos en forma de conjuntos de datos (*datasets*).
4. Se procederá al modelizado para la transformación de los datos en conocimiento, principalmente empleando técnicas de aprendizaje computacional.
5. Se validarán los resultados obtenidos con expertos en el dominio del problema para ver si coinciden con lo esperado.
6. Se difundirá el conocimiento en forma de publicaciones en revistas de alto impacto coherentes con la línea de investigación y en forma de software (aplicaciones de escritorio, aplicaciones web, scripts).

A continuación, se describe el tiempo estimado para la consecución de cada uno de los objetivos del proyecto de investigación que se propone como tesis para cada una de las fases de la metodología CRISP-DM.

1. *Herramienta_predicciones_energia_solar*
   a) Comprensión del problema. 1 mes

b) Comprensión de los datos. 1 mes
c) Preparación de los datos. 2 meses
d) Modelado. 4 meses
e) Evaluación. 1 mes
f) Difusión. 3 meses

2. *Herramienta_zonas_uhi*
a) Comprensión del problema. 1 mes
b) Comprensión de los datos. 1 mes
c) Preparación de los datos. 2 meses
d) Modelado. 4 meses
e) Evaluación. 1 mes
f) Difusión. 3 meses

3. *Herramienta_naturacion*
a) Comprensión del problema. 1/2 mes.
b) Comprensión de los datos. 1/2 mes.
c) Preparación de los datos. 1 mes.
d) Modelado. 2 meses.
e) Evaluación. 1 mes.
f) Difusión. 1 mes.

4. *Modelado_patrones_consumo_eléctrico*
a) Comprensión del problema. 1/2 mes.
b) Comprensión de los datos. 1/2 mes.
c) Preparación de los datos. 1 mes.
d) Modelado. 2 meses.
e) Evaluación. 1 mes.
f) Difusión. 1 mes.

La Figura 5 muestra un cronograma con la planificación prevista asociada a las tareas de cada fase del proceso de minería de datos relativas a la consecución de cada objetivo por alcanzar.

III. Relevancia

Las aportaciones derivadas de esta investigación que se pretenden ofrecer a la comunidad científica en forma de herramientas sencillas y gratuitas, y de publicaciones en revistas de alto impacto, servirán de ayuda en el proceso de toma de decisiones relevantes que puedan repercutir directamente en una notable mejora de la sostenibilidad urbana (realizar instalaciones verdes o fotovoltaicas, sustituir las energías contaminantes por energías renovables, mejorar la planificación de consumo eléctrico...)

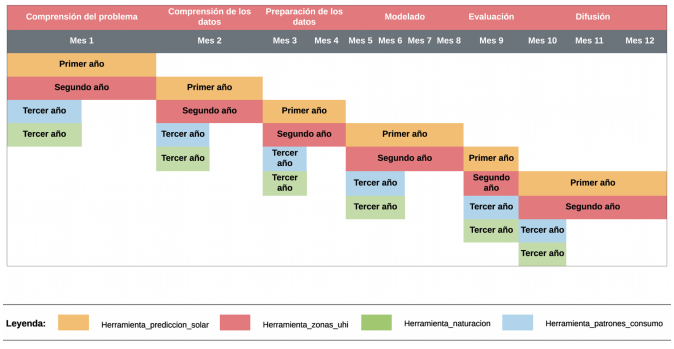


Figura 5. Planificación de tareas asociadas a los objetivos por alcanzar

Los resultados de la investigación podrán ser de gran utilidad a:

- *Expertos en el dominio:* Gracias a las herramientas que se pretenden aportar, los planificadores urbanos(urban planners) podrán conocer las posibles mejores ubicaciones dónde realizar instalaciones verdes o fotovoltaicas, lo que podría favorecer sustitución del uso de energías contaminantes por energías renovables, y el enverdecimiento de las ciudades para mejorar las zonas más desfavorecidas.
- *Compañías eléctricas:* Gracias a la herramienta que permita modelar el consumo eléctrico y clasificar a los consumidores en base a los patrones de consumo determinados, podrán ofrecerles tarifas y servicios personalizados, a la vez que podrán mejorar su gestión energética al poder predecir los consumos demandados por los clientes.
- *Consumidores:* Las herramientas podrán repercutir en un notable ahorro en las facturas gracias a los servicios y tarifas personalizadas que las compañías eléctricas podrán ofrecerle. También podrán aportar conocimiento acerca de la energía fotovoltaica a corto y largo plazo de la que dispondrían en caso de que decidieran instalar placas fotovoltaicas en las áreas urbanas en las que le interese realizar el estudio, pudiendo favorecer así la sustitución de energías contaminantes por energías renovables de los consumidores.

En cuanto a la difusión de conocimiento y resultados, se pretende realizar al menos 3 publicaciones en revistas científicas de alto impacto con temática relacionada a la línea de investigación propuesta, y tener disponibles de forma gratuita las herramientas para tratar los problemas descritos.

En el ámbito socio-económico se espera una mejor eficiencia energética en las ciudades que repercuta en un mayor ahorro, al mismo tiempo que ofrezca un mayor confort para los seres vivos. Si el estudio de localización y configuración de nuevas instalaciones (ya sean verdes o fotovoltaicas) motiva la instalación de sistemas fotovoltaicos y la naturación urbana, se estará contribuyendo a una mejor sostenibilidad urbana.

Por último, en cuanto a aspectos medioambientales, incrementará la mejora en la calidad de vida en las ciudades al reducir el efecto de isla de calor mediante el uso de las herramientas de detección de las zonas más desfavorables por el UHI y de las herramientas para la determinación de las posibles ubicaciones óptimas para la naturación que además ofrecerán las mejores formas de naturación para las superficies candidatas. Si por otro lado, también se consigue disminuir la demanda de energías contaminantes, estaremos ayudando a frenar el cambio climático a escala global.

IV. Agradecimientos

Este trabajo ha sido apoyado por el proyecto RTI2018-095097-B-I00 en la convocatoria 2018 del Proyecto I+D+i del Ministerio de Ciencia, Innovación Universidades, España.

REFERENCIAS

[1] B. S. Lin, C. C. Yu, A. T. Su, and Y. J. Lin, "Impact of climatic conditions on the thermal effectiveness of an extensive green roof," *Building and Environment*, vol. 67, pp. 26–33, 2013.

[2] Z. S. Venter, N. H. Krog, and D. N. Barton, "Linking green infrastructure to urban heat and human health risk mitigation in Oslo, Norway," *Science of the Total Environment*, vol. 709, p. 136193, 2020.

[3] J. S. Lee, J. T. Kim, and M. G. Lee, "Mitigation of urban heat island effect and greenroofs," *Indoor and Built Environment*, vol. 23, no. 1, pp. 62–69, 2 2014.

[4] T. Susca, S. Gaffin, and G. Dell'Osso, "Positive effects of vegetation: Urban heat island and green roofs," *Environmental Pollution*, vol. 159, no. 8-9, pp. 2119–2126, 8 2011.

[5] S. S. Herrera-Gomez, A. Quevedo-Nolasco, and L. Pérez-Urrestarazu, "The role of green roofs in climate change mitigation. A case study in Seville (Spain)," *Building and Environment*, vol. 123, pp. 575–584, 2017.

[6] H. Akbari and L. S. Rose, "Urban Surfaces and Heat Island Mitigation Potentials," *Journal of the Human-Environment System*, vol. 11, no. 2, pp. 85–101, 2008.

[7] R. C. Estoque, Y. Murayama, and S. W. Myint, "Effects of landscape composition and pattern on land surface temperature: An urban heat island study in the megacities of Southeast Asia," *Science of the Total Environment*, vol. 577, pp. 349–359, 2017.

[8] F. J. Fernández, L. J. Alvarez-Vázquez, N. García-Chan, A. Martínez, and M. E. Vázquez-Méndez, "Optimal location of green zones in metropolitan areas to control the urban heat island," *Journal of Computational and Applied Mathematics*, vol. 289, pp. 412–425, 2015.

[9] Y. Zhang, A. T. Murray, and B. L. Turner, "Optimizing green space locations to reduce daytime and nighttime urban heat island effects in Phoenix, Arizona," *Landscape and Urban Planning*, vol. 165, no. May, pp. 162–171, 2017.

[10] P. F. Jiménez-Pérez and L. Mora-López, "Modeling and forecasting hourly global solar radiation using clustering and classification techniques," *Solar Energy*, vol. 135, pp. 682–691, 2016.

[11] J. del Campo-Ávila, M. Piliougine, R. Morales-Bueno, and L. Mora-López, "A data mining system for predicting solar global spectral irradiance. Performance assessment in the spectral response ranges of thin-film photovoltaic modules," *Renewable Energy*, vol. 133, pp. 828–839, 4 2019.

[12] J. del Campo-Ávila, A. Takilalte, A. Bifet, and L. Mora-López, "Binding data mining and expert knowledge for one-day-ahead prediction of hourly global solar radiation," *Expert Systems with Applications*, p. 37.

[13] C. E. Kontokosta and C. Tull, "A data-driven predictive model of city-scale energy use in buildings," *Applied Energy*, vol. 197, pp. 303–317, 7 2017.

[14] R. Dutta, A. Morshed, J. Aryal, C. D'Este, and A. Das, "Development of an intelligent environmental knowledge system for sustainable agricultural decision support," *Environmental Modelling and Software*, vol. 52, pp. 264–272, 2 2014.

[15] J. R. Doorga, S. D. Rughooputh, and R. Boojhawon, "Multi-criteria GIS-based modelling technique for identifying potential solar farm sites: A case study in Mauritius," *Renewable Energy*, 2018.

[16] S. Fan, L. Chen, and W. J. Lee, "Machine learning based switching model for electricity load forecasting," *Energy Conversion and Management*, vol. 49, no. 6, pp. 1331–1344, 2008.

[17] J. del Campo-Ávila, G. Ramos-Jiménez, J. Gama, and R. Morales-Bueno, "Improving the performance of an incremental algorithm driven by error margins," *Intelligent Data Analysis*, vol. 12, no. 3, pp. 305–318, 2008.

[18] R. Moreno Sáez, M. Sidrach-De-Cardona, and L. Mora-López, "Data mining and statistical techniques for characterizing the performance of thin-film photovoltaic modules," *Expert Systems with Applications*, vol. 40, no. 17, pp. 7141–7150, 2013.

[19] R. Moreno-Sáez and L. Mora-López, "Modelling the distribution of solar spectral irradiance using data mining techniques," *Environmental Modelling and Software*, vol. 53, pp. 163–172, 2014.

[20] L. Kong, Z. Liu, and J. Wu, "A systematic review of big data-based urban sustainability research: State-of-the-science and future directions," *Journal of Cleaner Production*, vol. 273, 2020.

[21] S. Kiluk, "Diagnostic information system dynamics in the evaluation of machine learning algorithms for the supervision of energy efficiency of district heating-supplied buildings," *Energy Conversion and Management*, vol. 150, pp. 904–913, 10 2017.

[22] D. Toquica, K. Agbossou, R. Malhamé, N. Henao, S. Kelouwani, and A. Cardenas, "Adaptive machine learning for automated modeling of residential prosumer agents," *Energies*, vol. 13, no. 9, pp. 1–19, 2020.

[23] I. Frías-Blanco, J. del Campo-Ávila, G. Ramos-Jiménez, R. Morales-Bueno, A. Ortiz-Díaz, and Y. Caballero-Mota, "Online and non-parametric drift detection methods based on Hoeffding's bounds," *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, no. 3, pp. 810–823, 2015.

[24] I. Frías-Blanco, J. del Campo-Ávila, G. Ramos-Jiménez, A. C. Carvalho, A. Ortiz-Díaz, and R. Morales-Bueno, "Online adaptive decision trees based on concentration inequalities," *Knowledge-Based Systems*, vol. 104, pp. 179–194, 7 2016.

[25] y. B. T. . Baer, T., "The model development process. understand, manage, and prevent algorithmic bias," Tech. Rep.

[26] ——, "¿qué es el proceso de ciencia de datos en equipo (tdsp)?, and prevent algorithmic bias," Tech. Rep. [Online]. Available: https://docs.microsoft.com/es-es/azure/machine-learning/team-data-science-process/overview

[27] P. Chapman, J. Clinton, R. Kerber, T. Khabaza, T. Reinartz, C. Shearer, and R. Wirth, "Crisp-dm 1.0 step-by-step data mining guide," The CRISP-DM consortium, Tech. Rep., August 2000. [Online]. Available: https://maestria-datamining-2010.googlecode.com/svn-history/r282/trunk/dmct-teorica/tp1/CRISPWP-0800.pdf

PhD Thesis Proposal: Adversarial Machine Learning in the Audio Domain

Jon Vadillo
Intelligent Systems Group (ISG)
University of the Basque Country UPV/EHU
Donostia - San Sebastian, Spain
jon.vadillo@ehu.eus

Roberto Santana (Advisor)
Intelligent Systems Group (ISG)
University of the Basque Country UPV/EHU
Donostia - San Sebastian, Spain
roberto.santana@ehu.eus

Jose A. Lozano (Advisor)
Intelligent Systems Group (ISG)
University of the Basque Country UPV/EHU
Donostia - San Sebastian, Spain
ja.lozano@ehu.eus
Basque Center for Applied Mathematics (BCAM)
Bilbao, Spain
jlozano@bcamath.org

***Abstract*—Despite the extraordinary performance of Deep Neural Networks (DNNs) in a broad spectrum of artificial intelligence problems, it has been demonstrated that these models can be fooled by adversarial examples: inputs maliciously perturbed to drastically change the output of DNNs, whereas the perturbations are imperceptible to humans. The vulnerability of DNNs to adversarial examples poses a concerning security breach, especially in those scenarios in which security and reliability are critical factors. This thesis project aims to advance the field of adversarial machine learning, with a focus on the domain of intelligent audio processing, which has been considerably less studied than the image domain despite the growing use of audio recognition models in critical real-world applications. In particular, the objectives of this thesis are: (I) developing novel attack strategies in different audio tasks, (II) studying different properties of adversarial examples in order to shed light on why DNNs are vulnerable to them, (III) proposing defensive methods to make the models robust to audio adversarial perturbations, and (IV) assessing the applicability of the proposed approaches in real-world scenarios.**

***Index Terms*—adversarial examples, deep neural networks, machine learning, intelligent audio processing, speech recognition**

I. Introduction

Machine Learning models, and more particularly Deep Neural Networks (DNNs), are the state-of-the-art in a wide range of complex artificial intelligence problems, such as image classification, automatic speech recognition and robotics. Due to their remarkable performance, these models are currently deployed in multiple real-word applications where security is an utterly important requirement, such as autonomous driving, medical image analysis, malware classification or authentication systems. Nevertheless, it has been proven that DNNs can be fooled by an adversary. Precisely, it is possible to drastically change the output of a DNN, causing them to make wrong predictions, by altering the inputs with minimal perturbations that are imperceptible to humans. For instance, as illustrated in Figure 1, a traffic sign (e.g., a stop sign) can be modified so that image recognition models of self-driving cars will misclassify them (e.g., as a turn right sign) whereas the modifications will not raise any suspicion among human observers. These malicious inputs are known in the literature as *adversarial examples* [1], [2]. The possibility of generating such *adversarial attacks* implies a security breach in current technologies involving DNNs. Due to the implications, relevance, and scientific interest of adversarial examples, their study has become a hot-topic in Machine Learning, with multiple open and intensively studied lines of research. This thesis project addresses the main research directions, which will be overviewed along this section.

Most research efforts in the literature have been focused on developing new attack strategies. Overall, adversarial attacks can be classified into different groups according to their objectives or characteristics [3]. For instance, depending on the type of misclassification to be produced, adversarial attacks are *untargeted* when they are designed just to change the correct classification of the model, without any condition on the incorrect output class, and *targeted* when the aim is to produce a specific incorrect class. Adversarial attacks can be also grouped depending on the generality level of the perturbations. Most attack algorithms in the literature are based on generating *individual* perturbations, which consist of perturbations crafted to fool the model for one particular input [1], [2], [4]–[6]. Thus, it is not expected that the same perturbation will be effective if it is applied to different inputs.

This work is supported by the Basque Government (BERC 2018-2021 and ELKARTEK programs, IT1244-19, and PRE_2019_1_0128 predoctoral grant), by the Spanish Ministry of Economy and Competitiveness MINECO (projects TIN2016-78365-R and PID2019-104966GB-I00) and by the Spanish Ministry of Science, Innovation and Universities (FPU19/03231 predoctoral grant). Jose A. Lozano acknowledges support by the Spanish Ministry of Science, Innovation and Universities through BCAM Severo Ochoa accreditation (SEV-2017-0718).

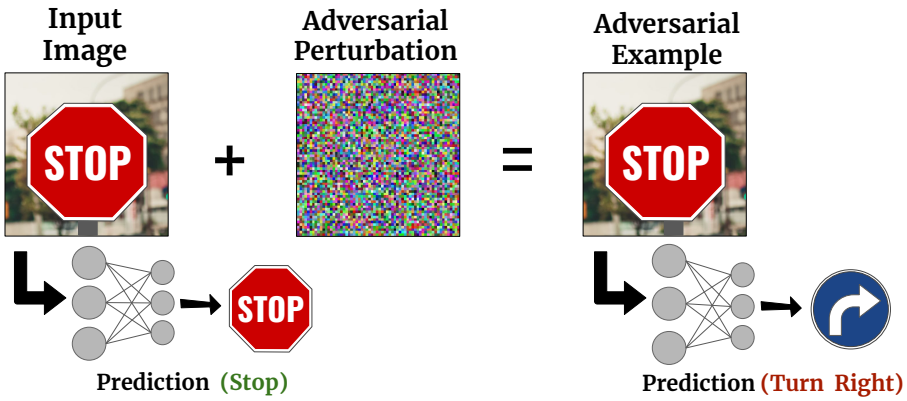


Fig. 1. Illustration of an adversarial attack.

However, it has been shown that input-agnostic perturbations can be also generated, known as *universal perturbations* [7], which are capable of fooling the model independently of the input to which they are applied. The intriguing fact that a single imperceptible perturbation is capable of fooling the model for multiple inputs is itself an important research line that we address in the project.

Generally, two common assumptions in the literature on offensive methods are that the adversary has a complete knowledge of the model to be attacked, information that can be used to craft the perturbations, and that the adversarial examples are generated *"digitally"*. However, many works have shown that adversarial examples can be generated in more challenging and realistic scenarios. For instance, some attack approaches are capable of fooling DNNs even in *black-box scenarios* in which no information about the model (e.g., architecture, parameters or training data) is available to the adversary [8], [9]. Other works focus instead on generating *robust* adversarial examples that can be deployed in the physical-world without losing the adversarial effect of the perturbation. Examples of physically-realizable attacks can be, among others, audio signals deployed over-the-air to trigger malicious actions in speech recognition systems [10], [11], or traffic signs which can fool image recognition systems at different distances, angles or even light conditions [12], [13]. The possibility of generating attacks under these conditions further emphasizes the concerning vulnerabilities of current DNN based systems. In this thesis, we aim to explore novel attack strategies capable of achieving more complex and ambitious objectives, providing new ways to maliciously control the output of DNNs.

Along with research on offensive methods, different defensive methods have been proposed to address these vulnerabilities [3], [14], which can be grouped into two categories. On the one hand, *proactive* strategies try to make the models more robust to adversarial examples so that it is more difficult to generate perturbations that mislead them [2], [15]–[17]. On the other hand, *reactive* strategies attempt to detect whether an input is adversarial or legit [18]–[21], or preprocess the inputs to remove potentially malicious perturbations in them before being sent to the DNN [22], [23]. Although multiple promising strategies have been proposed, it has been proven that most of them are ineffective or offer only limited effectiveness [15], [24], [25]. Moreover, most defensive methods are capable of increasing the robustness of the models only at the cost of sacrificing performance on clean (not perturbed) data [26]. Thus, there is still a lack of defensive mechanisms capable of guaranteeing a safe and reliable use of DNNs in adversarial scenarios.

Another related issue of great relevance is the lack of formal justifications capable of explaining the vulnerability of DNNs to adversarial examples. Different theoretical frameworks have been proposed in the literature, in which adversarial examples are attributed to intrinsic properties of DNNs [1], [2], to an insufficient amount of training data [27], or to the existence of specific features in the data itself [28]. Despite the numerous hypotheses, there is no consensus in the community. Similarly, several intriguing properties of adversarial examples are not well understood yet, such as their *transferability* [8], which implies that adversarial examples generated for a particular model are often effective against other models as well, which have not been considered during the attack generation process. However, due to the high-dimensionality of the decision spaces learned by DNNs and the black-box reasoning process of these models, studying their properties is often complex or even intractable, which hampers advances in these directions.

Finally, although the vulnerabilities to adversarial examples are present in all application domains, attack and defense methods have been mainly investigated in the image domain. Only recently has research been conducted in domains such as audio or text classification. The audio domain is particularly relevant due to the growing number of virtual agents, whose communication with humans is based on spoken commands, and where DNN based speech recognition systems play an important role. Therefore, advances in the study of adversarial attacks in the audio domain are a paramount priority. Besides, advances in multiple domains is also an important requirement in order to better understand the big picture of adversarial examples.

II. Objectives

The main objective of this thesis project is to advance the field of adversarial machine learning in the audio domain by exploring novel attack and defensive strategies, as well as by assessing their applicability in real-world applications. Ultimately, this research aims to contribute to raise awareness about the vulnerabilities of current audio classification models, develop more robust models, and get a better understanding of the vulnerability of DNNs to adversarial examples. For this purpose, the following specific objectives have been specified (summarized in Figure 2), and a detailed work plan has been developed oriented to achieve each of them (which will be described in Section III):

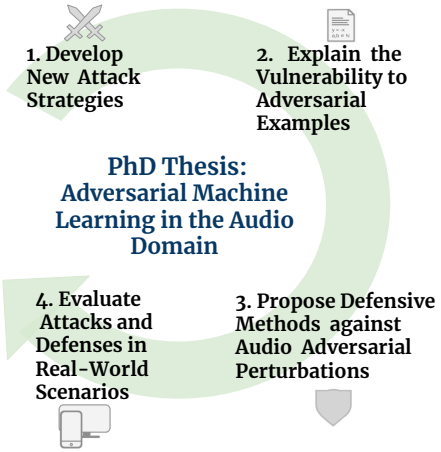


Fig. 2. Overview of the objectives of this PhD thesis.

O1 Develop new adversarial attack strategies to achieve more complex objectives. In this objective, we aim to generate novel adversarial attack strategies which, besides being able to fool a target model and ensuring that the perturbations are imperceptible, can be used to achieve novel or more complex goals. In particular, we identify the following sub-objectives:

1) **Generate audio perturbations with different levels of universality.** Previous works have focused on generating universal perturbations effective on any input, regardless of their input class. However, no previous work have explored the vulnerability of DNNs to audio perturbations designed to inhibit one class, or a subset of classes (e.g., fooling a DNN to misclassify some specific voice commands). Thus, we aim to study the feasibility of generating such perturbations in different audio tasks, contributing also to extend the notion of universal perturbations.
2) **Controlling the probability distribution of the output classes.** Current attack paradigms do not take into account the global effect of fooling a model for a large number of inputs. For instance, a change in the frequency with which each class is predicted can be a signal that the model is being attacked. For this reason, we aim to develop an attack strategy which, apart from fooling the model, is capable of controlling the probability with which the fooled DNN will (incorrectly) return each output class in the long run. This allows, among other applications, fooling the model for multiple inputs ensuring that the original (global) behavior of the model will remain unchanged, making the attacks less detectable, or forcing concept drifts [29] in online or adaptive learning scenarios.
3) **Generate adversarial examples transferable across different audio tasks.** One of the most relevant and intriguing properties of adversarial examples is their transferability, that is, their ability to fool different DNNs that have not been used during the generation of the attack. In this objective, we aim to take another step further and study whether adversarial examples generated for one audio task (e.g., speech command classification) can be also transferred across models trained for other tasks (e.g., sentiment analysis or speaker recognition).

Although these objectives are still in process, we have developed two scientific articles addressing the two first attack strategies [30], [31], which have been submitted for publication.

O2 Propose new hypotheses as well as theoretical and analytical frameworks to explain the vulnerability of DNNs to different adversarial attacks Studying why DNNs are vulnerable to adversarial perturbations is an important research direction to better understand the learning and reasoning process of these black-box models, and, ultimately, to correct their flaws. Thus, the second objective of this thesis will be to develop theoretical and analytical frameworks to rigorously study the properties of different adversarial perturbations, with special attention to universal and transferable perturbations, the attack approaches studied in the first objective. Although this objective is still in process, we have developed two scientific papers based on the findings of our research. In [32], which was published in an international conference, we introduce an analytical framework to study the level of accuracy with which adversarial attacks approximate the geometry of decision boundaries in high-dimensional problems. In [33], which is currently submitted for publication, we provide new insights on the vulnerability of speech recognition models to universal attacks.

O3 Propose effective defensive methods against audio adversarial perturbations. This objective aims to apply the knowledge gained in the previous phases to develop defensive methods, capable of making the DNNs more robust to audio adversarial attacks, which is of primary

importance to ensure a reliable use of Deep Learning models in safety-critical tasks. To achieve this objective, we identify as a promising research direction merging approaches proposed in the field of explainable Machine Learning with adversarial defense strategies. In fact, recent works in the image domain have shown that interpretability and adversarial robustness are connected, so that improving one of the properties can improve the other one as well [34], [35]. Thus, we hypothesize that the limitations of current defensive strategies can be overcome by adopting training paradigms in which classification performance, adversarial robustness and explainability are maximized jointly.

O4 Evaluate adversarial attack and defense strategies on real-world scenarios. Due to the rapid advances in voice-controlled intelligent agents during the last years, these technologies are being applied in multiple scenarios, many of which require high reliability. For example, speech based agents in smartphones, home-automation and other commercial tools are susceptible to be attacked, leading to privacy and cyber-security issues. Therefore, in this last part of the thesis project we aim to evaluate the proposed offensive and defensive methods in real-world scenarios, such as commercial speech recognition systems or models developed in real-world projects in which the ISG research group is currently collaborating.

III. Methodology and work plan

The general methodology that will be followed in order to achieve each of the objectives is as follows. First, once the related literature has been reviewed, a research plan has been determined and several preliminary hypotheses have been defined, efforts will be devoted to the achievement of the objectives through a combination of analytical work and experimental validation of the hypotheses through computational experiments. Once the necessary tasks have been carried out to achieve the objectives and to obtain the results of the research, an article will be written and submitted for consideration in a journal included in the ISI Journal Citation Report. In some cases, the preliminary results of the work will also be sent for consideration to prestigious conferences in the field.

The specific work plan that will be followed in order to fulfill each of the objectives defined in Section II is defined as follows:

O1 Develop new adversarial attack strategies to achieve more complex objectives.

1) Literature review of state-of-the-art adversarial attack strategies.
2) Identify a representative collection of datasets, related models and performance metrics with which the effectiveness of the attacks can be assessed depending on the tasks to be addressed or the objective to be achieved (e.g., universality or cross-task transferability).
3) Propose novel adversarial attack strategies which are capable of achieving the different objectives specified.
4) Evaluate the effectiveness of the proposed strategies according to different criteria, such as the fooling capability, the amount of distortion required and the specific performance metrics proposed for each particular objective.
5) Write a scientific paper describing the contributions derived from the work on the proposed objective.

O2 Propose new hypotheses as well as theoretical and analytical frameworks to explain the vulnerability of DNNs to different adversarial attacks

1) Review of the theoretical frameworks proposed in the literature to explain the existence of adversarial examples or their properties.
2) Critical analysis of the theoretical approaches described in the literature, identifying the application domains where their validity has been proven, the aspects of the algorithms' behavior that they are able to explain and also the gaps in the explanatory arguments provided in the literature.
3) Propose new hypotheses as well as theoretical and analytical frameworks to explain the vulnerability of DNNs to different adversarial attacks, with special attention to the study of properties such as universality or transferability.
4) Experimentally test the proposed frameworks in order to assess their validity.
5) Write a scientific paper describing the contributions derived from the work on the proposed objective.

O3 Propose effective defensive methods against audio adversarial perturbations.

1) Review of the literature on strategies proposed as defenses against adversarial attacks.
2) Identify the benefits and limitations of current state-of-the art defenses on the audio domain.
3) Propose new strategies to increase the adversarial robustness of audio classification models, with a special focus on integrating approaches from the field of explainable Machine Learning.
4) Evaluate the proposed defenses in terms of the robustness to multiple adversarial attacks types (e.g., individual, universal, transferred attacks, etc.) and compare the results with state-of-the-art approaches.
5) Write a scientific paper describing the contributions derived from the work on the proposed objective.

O4 Evaluate adversarial attack and defense strategies on real-world scenarios.

1) Review the literature on audio adversarial attacks

and defenses evaluated in real-world scenarios or applications (e.g., commercial speech recognition systems).

2) Identify representative real-world problems, prioritizing those on which the ISG research group is currently investigating, several of them in collaboration with leading companies in the field, such as Sherpa.ai, experts in the generation of intelligent assistants.
3) Apply the attack methods and defense strategies conceived as part of the previous objectives to the real-world problems identified, evaluating their effectiveness and their limitations.
4) Determine the additional challenges that real-world problems pose in comparison to *"virtual"* or *"digital"* scenarios, and propose solutions to overcome these challenges.
5) Write a scientific paper describing the contributions derived from the work on the proposed objective.

IV. Relevance

Machine Learning models are one of the most remarkable scientific advances of recent years, and their integration into real applications has proven to be of great value to society in multiple areas. Precisely because of their great social impact, it is essential to ensure that these models are safe, trustworthy and ethical, which is also a growing demand from multiple institutions and entities [36], [37]. We believe that this thesis may represent significant advances in the study of audio recognition models, whose vulnerabilities have not yet been sufficiently studied despite being one of the practical domains where DNNs are most used, due to their integration in smartphones, home-automation and voice-commanded agents.

On the other hand, advances in the field of adversarial examples can lead to advances in the more general field of Machine Learning, as both fields are deeply connected. For instance, research in adversarial attacks have contributed to advances in understanding how DNNs process the input samples along their architecture or how these models learn. We believe that the objectives defined in this thesis, such as studying the properties of different adversarial perturbations or defensive strategies, can bring along further insights about DNNs. Finally, the methods developed throughout the project will be applied to real problems involving intelligent audio processing models. We believe that the study of these systems beyond the academic standpoint or virtual scenarios can be of great utility and relevance, and that this research will contribute to establish the basis on how to develop more secure models.

References

[1] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing Properties of Neural Networks," in *International Conference on Learning Representations (ICLR)*, 2014, pp. 1–10.

[2] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and Harnessing Adversarial Examples," *arXiv preprint arXiv:1412.6572*, 2014.

[3] X. Yuan, P. He, Q. Zhu, and X. Li, "Adversarial Examples: Attacks and Defenses for Deep Learning," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 9, pp. 2805–2824, 2019.

[4] A. Kurakin, I. J. Goodfellow, and S. Bengio, "Adversarial Machine Learning at Scale," in *International Conference on Learning Representations (ICLR)*, 2017, pp. 1–17.

[5] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "DeepFool: A Simple and Accurate Method to Fool Deep Neural Networks," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 2574–2582.

[6] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards Deep Learning Models Resistant to Adversarial Attacks," in *International Conference on Learning Representations (ICLR)*, 2018, pp. 1–23.

[7] S.-M. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, and P. Frossard, "Universal Adversarial Perturbations," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 86–94.

[8] Y. Liu, X. Chen, C. Liu, and D. Song, "Delving into Transferable Adversarial Examples and Black-box Attacks," in *International Conference on Learning Representations (ICLR)*, 2017, pp. 1–14.

[9] N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. B. Celik, and A. Swami, "Practical Black-Box Attacks against Machine Learning," in *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*, ser. ASIA CCS '17, 2017, pp. 506–519.

[10] L. Schönherr, T. Eisenhofer, S. Zeiler, T. Holz, and D. Kolossa, "Imperio: Robust Over-the-Air Adversarial Examples for Automatic Speech Recognition Systems," in *Annual Computer Security Applications Conference*, ser. ACSAC '20, 2020, pp. 843–855.

[11] Y. Qin, N. Carlini, G. Cottrell, I. Goodfellow, and C. Raffel, "Imperceptible, robust, and targeted adversarial examples for automatic speech recognition," in *Proceedings of the 36th International Conference on Machine Learning*, vol. 97, 2019, pp. 5231–5240.

[12] A. Kurakin, I. Goodfellow, and S. Bengio, "Adversarial Examples in the Physical World," *arXiv preprint arXiv:1607.02533*, 2016.

[13] K. Eykholt, I. Evtimov, E. Fernandes, B. Li, A. Rahmati, C. Xiao, A. Prakash, T. Kohno, and D. Song, "Robust Physical-World Attacks on Deep Learning Visual Classification," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1625–1634.

[14] S. H. Silva and P. Najafirad, "Opportunities and Challenges in Deep Learning Adversarial Robustness: A Survey," *arXiv preprint arXiv:2007.00753*, 2020.

[15] F. Tramèr, A. Kurakin, N. Papernot, I. Goodfellow, D. Boneh, and P. McDaniel, "Ensemble Adversarial Training: Attacks and Defenses," in *International Conference on Learning Representations (ICLR)*, 2018, pp. 1–20.

[16] N. Papernot, P. McDaniel, X. Wu, S. Jha, and A. Swami, "Distillation as a Defense to Adversarial Perturbations against Deep Neural Networks," *arXiv preprint arXiv:1511.04508*, 2016.

[17] M. Abbasi and C. Gagné, "Robustness to Adversarial Examples through an Ensemble of Specialists," *arXiv preprint arXiv:1702.06856*, 2017.

[18] J. H. Metzen, T. Genewein, V. Fischer, and B. Bischoff, "On Detecting Adversarial Perturbations," in *International Conference on Learning Representations (ICLR)*, 2016, pp. 1–12.

[19] K. Grosse, P. Manoharan, N. Papernot, M. Backes, and P. McDaniel, "On the (Statistical) Detection of Adversarial Examples," *arXiv preprint arXiv:1702.06280*, 2017.

[20] K. Lee, K. Lee, H. Lee, and J. Shin, "A Simple Unified Framework for Detecting Out-of-Distribution Samples and Adversarial Attacks," in *Advances in Neural Information Processing Systems*, vol. 31, 2018, pp. 7167–7177.

[21] S. Bulusu, B. Kailkhura, B. Li, P. K. Varshney, and D. Song, "Anomalous Example Detection in Deep Learning: A Survey," *IEEE Access*, vol. 8, pp. 132 330–132 347, 2020.

[22] S. Gu and L. Rigazio, "Towards Deep Neural Network Architectures Robust to Adversarial Examples," *arXiv preprint arXiv:1412.5068*, 2015.

[23] D. Meng and H. Chen, "MagNet: A Two-Pronged Defense against Adversarial Examples," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '17, 2017, pp. 135–147.

[24] N. Carlini and D. Wagner, "Towards Evaluating the Robustness of Neural Networks," in *2017 IEEE Symposium on Security and Privacy (SP)*, 2017, pp. 39–57.

[25] A. Athalye, N. Carlini, and D. Wagner, "Obfuscated Gradients Give a False Sense of Security: Circumventing Defenses to Adversarial

Examples," in *Proceedings of the 35th International Conference on Machine Learning*, 2018, pp. 274–283.
[26] H. Zhang, Y. Yu, J. Jiao, E. Xing, L. E. Ghaoui, and M. Jordan, "Theoretically Principled Trade-off between Robustness and Accuracy," in *Proceedings of the 36th International Conference on Machine Learning*, 2019, pp. 7472–7482.
[27] L. Schmidt, S. Santurkar, D. Tsipras, K. Talwar, and A. Madry, "Adversarially Robust Generalization Requires More Data," in *Advances in Neural Information Processing Systems*, vol. 31, 2018, pp. 5014–5026.
[28] A. Ilyas, S. Santurkar, D. Tsipras, L. Engstrom, B. Tran, and A. Madry, "Adversarial Examples Are Not Bugs, They Are Features," in *Advances in Neural Information Processing Systems 32*, 2019, pp. 125–136.
[29] J. Gama, I. Žliobaitė, A. Bifet, M. Pechenizkiy, and A. Bouchachia, "A Survey on Concept Drift Adaptation," *ACM Computing Surveys*, vol. 46, no. 4, pp. 44:1–44:37, 2014.
[30] J. Vadillo and R. Santana, "Universal Adversarial Examples in Speech Command Classification," *arXiv preprint arXiv:1911.10182*, 2019, submitted for publication.
[31] J. Vadillo, R. Santana, and J. A. Lozano, "Extending Adversarial Attacks to Produce Adversarial Class Probability Distributions," *arXiv preprint arXiv:2004.06383*, 2020, submitted for publication.
[32] ——, "Exploring Gaps in DeepFool in Search of More Effective Adversarial Perturbations," in *Machine Learning, Optimization, and Data Science*, ser. Lecture Notes in Computer Science, 2020, pp. 215–227.
[33] ——, "Analysis of Dominant Classes in Universal Adversarial Perturbations," *arXiv preprint arXiv:2012.14352*, 2021, submitted for publication.
[34] A. Boopathy, S. Liu, G. Zhang, C. Liu, P.-Y. Chen, S. Chang, and L. Daniel, "Proper Network Interpretability Helps Adversarial Robustness in Classification," in *Proceedings of the 37th International Conference on Machine Learning*, vol. 119, 2020, pp. 1014–1023.
[35] A. Noack, I. Ahern, D. Dou, and B. Li, "An Empirical Study on the Relation Between Network Interpretability and Adversarial Robustness," *SN Computer Science*, vol. 2, no. 1, p. 32, 2021.
[36] B. Goodman and S. Flaxman, "European Union Regulations on Algorithmic Decision-Making and a 'Right to Explanation'," *AI Magazine*, vol. 38, no. 3, pp. 50–57, 2017.
[37] C. Rudin, "Stop Explaining Black Box Machine Learning Models for High Stakes Decisions and Use Interpretable Models Instead," *Nature Machine Intelligence*, vol. 1, no. 5, pp. 206–215, 2019.

Artificial intelligence techniques for improving the perception systems of autonomous vehicles

Manuel Carranza García*, Jorge García-Gutiérrez†, José C. Riquelme†
Division of Computer Science, University of Seville
Seville, Spain
*mcarranzag@us.es
†Advisors: jorgarcia@us.es, riquelme@us.es

Starting date: 23 September 2019

***Abstract*—Autonomous driving has the potential to revolutionize mobility, but still presents many challenges that need to be addressed. The correct perception of the environment using remote sensing data is one of the fundamental aspects in which improvements are required. In this thesis project, our aim is to enhance the perception systems of autonomous vehicles, addressing the object detection task from different perspectives. Recent works in 2D object detection are mainly focused on the general-purpose COCO benchmark. However, their use in other domains such as autonomous driving, in which real-time inference is required, is yet an area to be fully explored. After performing an experimental revision of the speed/accuracy trade-off of existing deep learning-based 2D object detectors, our goal is to design a detection system that is adapted for this scenario. The proposed detector considers aspects such as the perspective projection of the vehicle's cameras and the high imbalance between different instance classes, while also meeting the efficiency requirements. Furthermore, we aim to develop temporal methods that incorporate information from past frames to improve detection precision. Then, the last objective is to increase the robustness of the detector by exploring data fusion techniques between 2D camera and 3D LiDAR data. The main data resource for the project will be the Waymo Open Dataset, which is the most extensive and diverse multi-modal self-driving dataset up to date.**

***Index Terms*—autonomous driving, convolutional neural networks, data fusion, deep learning, object detection, perception**

I. Introduction

Autonomous vehicles are increasingly earning popularity due to their potential to improve road safety and reduce environmental pollution. However, significant progress is still required until the Advanced Driver-Assistance System (ADAS) of these vehicles can fully substitute human driving. Among the many tasks that the ADAS needs to address, the correct perception of the environment is fundamental. Very recently, many self-driving car companies have publicly released high-quality perception datasets [1]. After the popular KITTI benchmark [2], there have been perception challenges organized with several larger-scale datasets such as PandaSet [3], NuScenes [4], and Google Waymo [5]. The fact that companies are making their data available to the research community indicates that there is still the need for pushing the state-of-the-art in many perception problems.



Fig. 1. Object detection for autonomous driving

This thesis project is mainly focused on the object detection task using the data from the on-board sensors of autonomous vehicles. This detection problem consists on determining the presence of objects from given categories and return the spatial location of each instance through a bounding box. An example is shown in Figure 1. With the increase of availability and quality of remote sensing data collected by multi-modal sensors (high resolution RGB cameras, LiDAR and radar data, etc.), deep learning models have become the reference approach for object detection [6]. Nevertheless, in the context of autonomous driving, it is still challenging for existing detectors to operate robustly in complex scenarios such as mixed urban traffic and adverse weather. Furthermore, the results obtained in recent detection challenges, such as the one hosted by Waymo, demonstrate that improvements are needed in order to perceive all traffic participants accurately and in real-time [7].

Given this background, the goal of this project is to enhance the perception systems of autonomous vehicles by improving the performance over the 2D object detection task. We propose a project that is divided into four stages, addressing the detection problem from different perspectives. Firstly, we aim to assess the performance of existing generic object detectors

within this particular context. The second objective is to design a detector that is specifically adapted for this scenario, which proposes a novel anchor optimization method and alleviates the class imbalance problem. Thirdly, we plan to develop temporal object detection methods to leverage the sequential nature of data generated by autonomous vehicles. Finally, we aim to explore techniques to fuse data from camera and LiDAR sensors in order to improve the robustness of the proposed detection system.

Recent works in the 2D object detection literature have excessively focused on the general-purpose COCO dataset [8], which has become the standard benchmark to validate novel proposals. However, the capacity of generalization of these methods to other datasets and domains is often unknown and the performance can differ significantly [9]. Furthermore, in many studies, the analysis is mainly based on outperforming the accuracy benchmarks, while ignoring the computational efficiency. In real-time applications such as autonomous driving, the required computational resources and the inference speed are essential aspects to consider. With the many existing architectures, it is difficult for practitioners to find out which is the most suitable for this scenario. For all these reasons, the community would benefit from experimental reviews on object detection architectures for autonomous driving.

Moreover, the default configuration of these generic detection architectures is not optimal for the particular characteristics of the vehicles' cameras. Existing pre-trained models in popular frameworks such as MMDetection [10] or Detectron2 [11] are mostly designed for the objects scales within the COCO dataset. However, the objects in the autonomous driving scenario present a significantly different distribution, which requires the models to be adapted in order to achieve superior performance [12]. Moreover, in existing self-driving datasets, the vehicle class is overrepresented compared to the number of instances from other classes such as pedestrians or cyclists. This imbalance problems causes overfitting issues and prevents from obtaining high precision over minority classes. Despite these issues, there are still few works that have addressed the optimization of 2D object detectors for autonomous driving, and most of them have only focused on vehicle detection [13]–[15].

Another interesting aspect that this project aims to cover is to incorporate the temporal component into the detection system. Most research on object detection has been constrained to processing individual camera frames. However, there have been recent works that have demonstrated the importance of temporal memory for building accurate video detectors [16], [17]. In newly large self-driving datasets such as Waymo, the data is recorded in series of 20 seconds, which provides a unique opportunity to develop methods that leverage the temporal information between consecutive frames.

The last part of the project focuses on multi-sensor data fusion approaches, taking advantage of the synchronized LiDAR and camera data provided in existing datasets. Relying only on camera data may not be sufficient in complex driving situations or at night. On the other hand, using only laser range data makes the detection task very challenging due to the limited density of the scans and the absence of appearance information. Therefore, there is the need for designing efficient data fusion methods that can outperform single 2D object detectors, while operating at real-time speed. Eventually, fusing information from multiple sensors across time would provide a more reliable detection system. Most of the existing works on this path have proposed fusion methods for 3D detection [18]. In contrast, 2D detection combining images and LiDAR projections, which normally requires less computation, has received less attention [19].

The next sections present the specific objectives, the methodology and the relevance of the proposed project. Furthermore, the first results obtained are briefly reported.

II. Objectives

The main objective of this thesis is to improve the perception systems of autonomous vehicles, focusing on the 2D object detection task. The goal is to design more robust and accurate deep learning-based systems that can provide real-time detections under diverse and complex driving situations. The main data resource to be used will be the Waymo Open Dataset, which is the most extensive self-driving dataset up to date. The project can be divided into the following four sub-objectives:

1) **Analysis of the performance of the existing 2D object detection systems in the literature.**
   As stated in the introduction, there is an overwhelming number of detection architectures that have been proposed in the literature. However, these detection frameworks have mainly been evaluated over the COCO dataset and their use in other specific applications has not been fully explored. Therefore, the first sub-objective is to review the state-of-the-art and evaluate the performance of popular deep learning 2D detectors in the context of autonomous vehicles. For this purpose, we will revise the suitability of different existing detection toolboxes that can be used. The idea is to use the COCO pre-trained models from these repositories in order to do transfer learning, which would allow evaluating how well these models adapt to our particular task.
   The final goal is to analyze the trade-off between accuracy and efficiency of detectors in this domain, in which providing real-time detections is essential. The performance analysis will include many metrics such as the global and per-class detection precision, inference speed, memory usage, and the number of parameters and floating-point operations. It is particularly interesting to see how the models perform over the three different classes that have to be detected: vehicles, pedestrians and cyclists.
   This objective has already been completed, having published an experimental review with the obtained findings in the Remote Sensing journal [20]. This study provides a general overview on which models are the most suitable given different settings and requirements.

2) **Proposal of an enhanced 2D object detector adapted for autonomous driving**
The second sub-objective can be considered the core of the project, and consists on developing an improved detector that is better suited for this domain. The idea is to propose several enhancements to the popular two-stage Faster R-CNN convolutional detector, which obtained the best speed/accuracy trade off in the experimental review [20]. Two main aspects are considered for increasing the performance over the original baseline: improving the anchor generation procedure and alleviating the performance drop in minority classes.
The default uniform anchor configuration of Faster R-CNN is not suitable in this scenario due to the perspective projection of the vehicle's cameras. In the images captured by on-board cameras, there is a strong positive correlation between the objects' scales and their position in the image. This implies that larger objects tend to be in the lower-part of the image and smaller objects appear at the upper-part. This correlation does not exist in the COCO dataset, hence default anchoring does not account for it. Therefore, the first goal is to propose a perspective-aware anchor optimization methodology that provides a better match to the objects' scales and shapes. We aim to achieve this by using evolutionary algorithms to optimize the configuration of different regions of the image that have to be defined.
Furthermore, the class imbalance problem also needs to be addressed. For instance, in the Waymo dataset, there is a high imbalance in the number of instances of pedestrians and cyclists compared to the large number of vehicles. Therefore, the goal is to propose modifications to the traditional loss function of Faster R-CNN that can improve the precision over the under-represented classes. Another interesting approach is to design ensemble models that combine the predictions of detectors that are diversely trained.
Our previous experience on remote sensing data classification using convolutional neural networks [21] has been helpful for approaching this objective successfully. We have recently published this enhanced 2D object detector for autonomous driving in the Neurocomputing journal, obtaining a significant improvement compared to the original model [12].

3) **Temporal methods for object detection**
The third sub-objective is to propose methods that leverage the temporal information naturally present in the data generated from autonomous vehicles. Since on-board sensors are continuously gathering data, it seems natural to develop solutions that use the knowledge obtained from previous frames to predict the current one. However, despite being critical for autonomous driving, video object detection has received less attention in the literature than still images.
The goal is to propose a modification to the enhanced 2D object detector, designed in the previous objective, in order to deal with sequential camera frames in a recurrent fashion. The proposal should also focus on building a low-latency video detector, which can operate in a real-world environment.
We are currently starting to pursue this objective, and a research stay at the University of California Berkeley will be carried out this year to work on it. The stay, that is sponsored by a Fulbright fellowship, will be with professor Avide Zakhor, who actively works in projects of the Berkeley Deep Drive (BDD) center related to autonomous driving using temporal LiDAR data [22].
The fact that our research group already has experience in problems with time series data further supports the suitability of this objective. In parallel to this thesis project, we have been working on time series forecasting problems from different domains, obtaining four related publications in leading journals such as IJNS or ICAE [23]–[26]. In these studies, we have developed deep learning architectures such as recurrent and temporal convolutional networks for both streaming and offline tasks. The knowledge acquired from these studies can be very useful for the development of detection architectures that exploit the temporal aspects of sequential camera images.
The work carried out in [24] is particularly interesting for this project. We designed an asynchronous dual-pipeline framework that allows using complex deep learning architectures for classifying data in streaming. This framework could also be used in our project, having a streaming detector that offers real-time inference, while also being constantly updated to changes in the data distribution.

4) **Multi-sensor data fusion**
The last sub-objective is to increase the robustness of the proposed detector by fusing data from different modalities. Under adverse weather conditions or at night, images from the RGB cameras may not be sufficient to perceive all traffic participants. Therefore, it seems necessary to study how to fuse data from the 2D camera and the 3D LiDAR sensors efficiently. The goal is to combine the 2D images with the associated LiDAR projections to leverage the advantages of both modalities and compensate for the defect of individual sensors. The idea is to study different data fusion architectures that could be integrated into the proposed detection system without a significant computational overhead. For this purpose, we would need to develop a preprocessing methodology to transform the sparse LiDAR projection into dense maps, which could be used as extra input channels in the deep neural network to enhance detection accuracy. Incorporating this depth information from laser scans would allow the detections to be more reliable in complex driving situations, hence improving the level of safety of the vehicle.

III. Methodology and work plan

This section presents the methodology to be followed in order to achieve the proposed objectives. Given that the project is expected to be completed in four years, the proposal is to dedicate one year for each of the four sub-objectives. Figure 2 presents a general overview of the work plan, highlighting which items have already been addressed and the current stage of the project.

1) **Analysis of the performance of existing 2D object detectors for autonomous driving (first year):**
   - 1.1 Comprehensive revision of existing generic deep learning meta-architectures and convolutional feature extractors backbones for 2D object detection.
   - 1.2 Revision of existing software toolboxes for object detection to determine the most convenient to be used.
   - 1.3 Conduct an experimental review using the Waymo Open Dataset. This review will analyze the speed/accuracy trade-off between one-stage and two-stage meta-architectures in combination with different backbones and image resolutions. The performance will be assessed over different metrics such as the per-class detection precision (vehicles, pedestrians and cyclists), inference speed, and memory usage.
   - 1.4 Submission of the experimental review with the obtained conclusions to a JCR journal. This study has already been published in Remote Sensing.
2) **Enhanced 2D object detector for autonomous driving (second year):**
   - 2.1 Design a perspective-aware anchor optimization methodology, based on evolutionary algorithms, to improve the performance of the default anchor configuration.
   - 2.2 Address the class imbalance problem with two different approaches. Firstly, study the use of different loss functions to alleviate the foreground-foreground imbalance. Secondly, design an ensemble model to increase the capacity of generalization of the models.
   - 2.3 Participation in the Waymo Open Dataset 2D object detection challenge to validate the effectiveness of our proposal.
   - 2.4 Writing of a paper presenting the proposed methodology and submission to a JCR journal. This work has already been published in the Neurocomputing journal.
3) **Temporal methods for object detection (third year)**
   - 3.1 In parallel to this project, carry out research on deep learning for time series forecasting/classification problems, both offline and in streaming. Four related studies have already been published.
   - 3.2 Research stay at University of California Berkeley with Professor Avide Zakhor to work on temporal methods for autonomous driving.
   - 3.3 Modify the enhanced detector to allow introducing temporal information for detecting objects over sequential images.
   - 3.4 Writing of a paper presenting the proposed methodology and submission to a JCR journal.
4) **Multi-sensor data fusion (fourth year):**
   - 4.1 Revision of existing literature on deep learning-based data fusion techniques.
   - 4.2 Design a procedure to preprocess sparse LiDAR projections to create 2D dense depth maps with information from the laser scans.
   - 4.3 Modify the enhanced detector to combine data from RGB images and depth maps.
   - 4.4 Writing of a paper presenting the proposed methodology and submission to a JCR journal.

IV. Relevance

Autonomous vehicles will surely produce revolutionary changes to the way we understand mobility in the future. They have the potential to reduce the use of fossil fuels and the transportations costs, as well as increasing safety in the road. % of road accidents are caused by human errors, hence self-driving cars can potentially save many lives. However, the scientific community is still struggling to solve the many challenges that autonomous driving presents. Among those challenges, perception is a fundamental area in which improvements are needed, and object detection is a core task.

Although there are many big technological companies trying to develop autonomous driving solutions, detecting all traffic participants accurately remains an unsolved problem. These companies, such as Google or Lyft, have recently shared their data so that researchers from all over the world can contribute to the development of driverless cars. These multi-modal large-scale datasets provide a unique opportunity to work on improving existing object detection architectures. They also allow approaching the problem from different perspectives than can be complemented: diverse driving situations, sequential and multi-modal data from different sensors, latency requirements, etc. The release of these data is a call for the community to speed up the process of autonomous driving becoming a reality. The impact that this technology can have for society is enormous, and it was the main reason for embarking ourselves into this project.

This project will contribute to build more robust and efficient detection systems that improve the state-of-the-art. The final goal is that our proposals could be applied into real-world systems to detect object and avoid collisions more accurately. Generic object detectors fail to provide optimal performance, hence there is the need for more specialized models that meet the specific requirements of this context. Given that

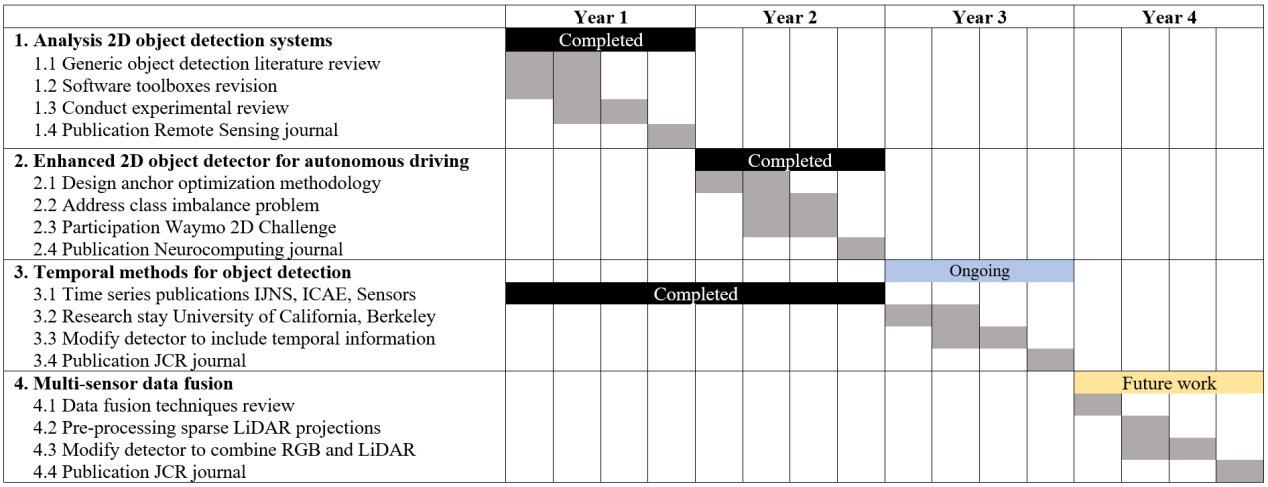


Fig. 2. Research plan for the project

autonomous driving is receiving increasing attention, building innovative solutions within this domain and transferring knowledge to society is of utmost importance.

V. First results

During the first two years of this project, we have successfully addressed the two first objectives that were described in Section II. For both studies, we have used the large Google Waymo Open Dataset, which has more than a million images in diverse geographies and conditions.

For the first objective, we performed an experimental review on object detection architectures for autonomous driving [20]. In this study, we aimed to evaluate the performance of existing deep learning models on a multi-class (vehicles, pedestrians, and cyclists) 2D detection problem. We performed experiments with several one-stage (RetinaNet, FCOS, and YOLOv3) and two-stage (Faster R-CNN) meta-architectures under different image resolutions and feature extractors (ResNet, ResNeXt, Res2Net, DarkNet, and MobileNet). Our findings demonstrated that two-stage detectors still provide the most robust and efficient performance. Faster R-CNN models outperform one-stage detectors in accuracy, achieving superior precision in the detection of underrepresented classes. The best speed/accuracy trade-off was obtained with Faster R-CNN Res2Net-101, but lower resolution images were needed to reach real-time speed. Furthermore, the anchor-free FCOS detector proved to be slightly faster than RetinaNet, with similar precision and lower memory usage.

For the second objective, we developed an enhanced 2D object detector based on Faster R-CNN that is better suited for the context of autonomous vehicles [12]. Our proposal improved two aspects: the anchor generation procedure, and the performance drop in minority classes. To optimize the anchor configuration, we proposed a perspective-aware methodology. The method divides the image into key regions via clustering and uses evolutionary algorithms to find the optimal base anchors for each region. Furthermore, we added a module that includes the spatial information of the candidate regions proposed in the first stage for the final box regression. The proposed architecture is shown in Figure 3. Additionally, we explored several re-weighting strategies to address the foreground-foreground class imbalance. We found out that a reduced version of focal loss can significantly improve the precision over difficult and minority instances in two-stage detectors. The last contribution was to design an ensemble model that combines the different learning strategies. With our proposal, we obtained an improvement over the original baseline of 6.13% mAP when using the best single model, and of 9.69% mAP with the ensemble. The proposed modifications over the Faster R-CNN can be extended to other anchor-based architectures and do not introduce additional computational cost at inference.

Regarding the third objective, we have been working in parallel in other projects such as the ADLStream framework [24]. In this work, we proposed a dual-pipeline architecture to allow using deep learning models in streaming. We proposed a system in which the data stream is dual fed into two different layers in order to concurrently provide quick predictions and update the deep learning model. This separation reduces processing time while obtaining high accuracy on classification. The proposed framework can be used with any kind of neural network, hence it could be applied also for this project in order to detect objects in streaming.

Acknowledgment

This research is supported by FEDER/Ministerio de Ciencia, Innovación y Universidades – Agencia Estatal de Investigación/Proyecto TIN2017-88209-C2 and by the Andalusian Regional Government under the projects: BIDASGRI: Big Data technologies for Smart Grids (US-1263341), Adaptive hybrid models to predict solar and wind renewable energy production (P18-RT-2778). M. Carranza-García is also supported by the predoctoral grant FPU18/00622.

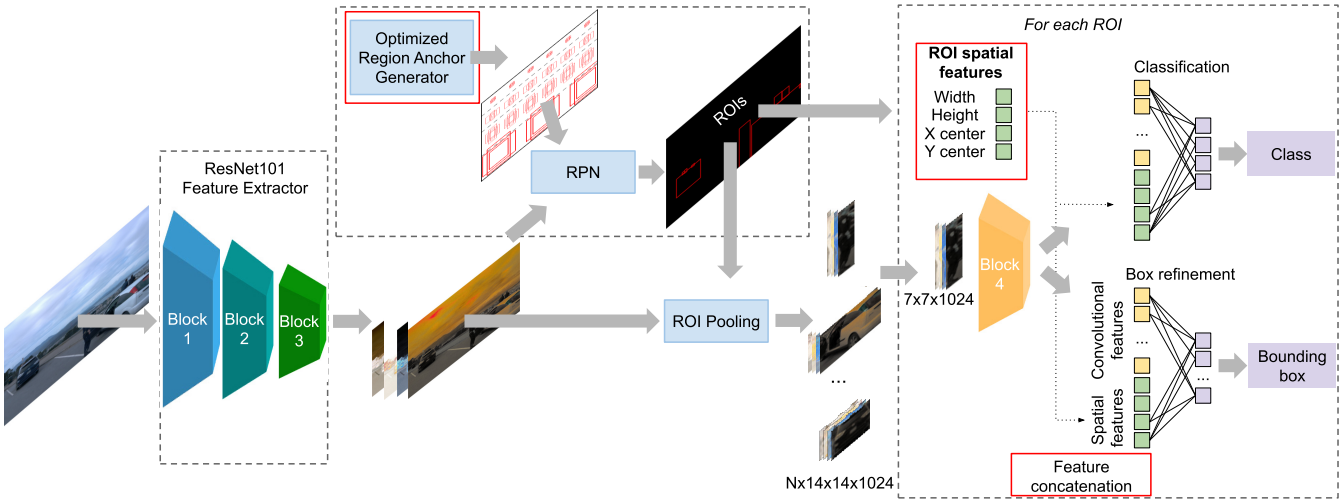


Fig. 3. Enhanced Faster-RCNN detector architecture for autonomous driving.

REFERENCES

[1] D. Feng *et al.*, "Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–20, 2020.

[2] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 3354–3361.

[3] H. . Scale, "PandaSet: Public large-scale dataset for autonomous driving," Available online: https://scale.com/open-datasets/pandaset, 2019, (Accessed 6 June 2021).

[4] H. Caesar *et al.*, "nuscenes: A multimodal dataset for autonomous driving," *CoRR*, vol. abs/1903.11027, 2019.

[5] P. Sun *et al.*, "Scalability in Perception for Autonomous Driving: Waymo Open Dataset," *CoRR*, vol. abs/1912.04838, 2019.

[6] L. Liu *et al.*, "Deep learning for generic object detection: A survey," *International Journal of Computer Vision*, vol. 128, no. 2, pp. 261–318, 2020.

[7] "Scalability in Autonomous Driving Workshop CVPR 2020," Available online: https://sites.google.com/view/cvpr20-scalability, 2020, (Accessed 8 June 2021).

[8] T.-Y. Lin *et al.*, "Microsoft COCO: Common Objects in Context," in *Computer Vision – ECCV 2014*. Cham: Springer International Publishing, 2014, pp. 740–755.

[9] Y. Shinya, "USB: universal-scale object detection benchmark," *CoRR*, vol. abs/2103.14027, 2021.

[10] K. Chen *et al.*, "MMDetection: Open mmlab detection toolbox and benchmark," *CoRR*, vol. abs/1906.07155, 2019. [Online]. Available: http://arxiv.org/abs/1906.07155

[11] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, and R. Girshick, "Detectron2," Available online: https://github.com/facebookresearch/detectron2, 2019, (Accessed 14 December 2020).

[12] M. Carranza-García, P. Lara-Benítez, J. García-Gutiérrez, and J. C. Riquelme, "Enhancing object detection for autonomous driving by optimizing anchor generation and addressing class imbalance," *Neurocomputing*, vol. 449, pp. 229–244, 2021.

[13] R. N. Rajaram, E. Ohn-Bar, and M. M. Trivedi, "RefineNet: Refining Object Detectors for Autonomous Driving," *IEEE Transactions on Intelligent Vehicles*, vol. 1, no. 4, pp. 358–368, Dec 2016.

[14] Y. Wang, Z. Liu, and W. Deng, "Anchor Generation Optimization and Region of Interest Assignment for Vehicle Detection," *Sensors*, vol. 19, p. 1089, 03 2019.

[15] M. Hassaballah, M. A. Kenk, K. Muhammad, and S. Minaee, "Vehicle detection and tracking in adverse weather using a deep learning framework," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–13, 2020.

[16] M. Liu, M. Zhu, M. White, Y. Li, and D. Kalenichenko, "Looking fast and slow: Memory-guided mobile video object detection," *CoRR*, vol. abs/1903.10172, 2019.

[17] D. Liu, Y. Cui, Y. Chen, J. Zhang, and B. Fan, "Video object detection for autonomous driving: Motion-aid feature calibration," *Neurocomputing*, vol. 409, pp. 1–11, 2020.

[18] X. Chen, H. Ma, J. Wan, B. Li, and T. Xia, "Multi-view 3d object detection network for autonomous driving," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 6526–6534.

[19] K. Geng, G. Dong, G. Yin, and J. Hu, "Deep dual-modal traffic objects instance segmentation method using camera and lidar data for autonomous driving," *Remote Sensing*, vol. 12, no. 20, 2020.

[20] M. Carranza-García, J. Torres-Mateo, P. Lara-Benítez, and J. García-Gutiérrez, "On the performance of one-stage and two-stage object detectors in autonomous vehicles using camera data," *Remote Sensing*, vol. 13, no. 1, 2021.

[21] M. Carranza-García, J. García-Gutiérrez, and J. C. Riquelme, "A Framework for Evaluating Land Use and Land Cover Classification Using Convolutional Neural Networks," *Remote Sensing*, vol. 11, no. 3, p. 274, 2019.

[22] S. McCrae and A. Zakhor, "3d object detection for autonomous driving using temporal lidar data," *Proceedings - International Conference on Image Processing, ICIP*, pp. 2661–2665, 2020.

[23] P. Lara-Benítez, M. Carranza-García, J. M. Luna-Romera, and J. C. Riquelme, "Temporal convolutional networks applied to energy-related time series forecasting," *Applied Sciences*, vol. 10, no. 7, 2020.

[24] P. Lara-Benítez, M. Carranza-García, J. García-Gutiérrez, and J. Riquelme, "Asynchronous dual-pipeline deep learning framework for online data stream classification," *Integrated Computer-Aided Engineering*, vol. 27, no. 2, pp. 101–119, 2020.

[25] P. Lara-Benítez, M. Carranza-García, F. Martínez-Álvarez, and J. C. Riquelme, "On the performance of deep learning models for time series classification in streaming," in *15th International Conference on Soft Computing Models in Industrial and Environmental Applications (SOCO 2020)*. Springer International Publishing, 2021, pp. 144–154.

[26] P. Lara-Benítez, M. Carranza-García, and J. C. Riquelme, "An experimental review on deep learning architectures for time series forecasting," *International Journal of Neural Systems*, vol. 31, no. 03, p. 2130001, 2021.

Design and Implementation of Metaheuristic Algorithms for Social Network Analysis

Sergio Pérez-Peló
Dept. Computer Science
Universidad Rey Juan Carlos
C/Tulipán, S/N,Móstoles, Spain
sergio.perez.pelo@urjc.es

Jesús Sánchez-Oro
Dept. Computer Science
Universidad Rey Juan Carlos
C/Tulipán, S/N,Móstoles, Spain
jesus.sanchezoro@urjc.es

Abraham Duarte
Dept. Computer Science
Universidad Rey Juan Carlos
C/Tulipán, S/N,Móstoles, Spain
abraham.duarte@urjc.es

***Abstract*—This Doctoral Thesis project, started in 2018, has the main objective of proposing new metaheuristic algorithms for dealing with several combinatorial optimization problems related to social network analysis and, in particular, with the detection of communities in social networks. The problem of community detection is becoming one of the key tasks in social network analysis, since it allows to analyze groups of users with similar interests. The resulting analysis is a key aspect in real applications of great relevance, such as radicalism detection or the reduction of the amount of data we have to process, among other applications. This is a difficult problem to solve, for which it is not possible to design an algorithm that finds the optimal solution in a reasonable computation time. If, in addition to this characteristic, we take into account the size that social networks tends to present nowadays, the result is a problem with a high number of applications for which no efficient solution is available. In this context, it is interesting to develop heuristic algorithms that allow to address the problem efficiently, obtaining high quality solutions, with a lower computational cost than exact algorithms. In this thesis competitive algorithms with the state-of-the-art methods are developed. These algorithms solve several optimization problems related to the family of community detection problems in social networks from a heuristic point of view.**

***Index Terms*—Heuristics, Metaheuristics, Community Detection, Social Network Analysis**

PhD. Candidate Information

- **Name**: Sergio Pérez-Peló
- **Supervisors**: Abraham Duarte and Jesús Sáncchez-Oro
- **Department**: Dept. of Computer Science, Universidad Rey Juan Carlos
- **Starting Date**: September, 2018

I. Introduction

The evolution of social networks in the last decades has aroused the interest of scientists from different fields, from psychology to computer science. Millions of people are constantly sharing all their personal and professional information on different social networks [1]. In addition, social networks have become one of the most widely used sources of information, mainly due to their ability to provide the user with real-time content. Social networks are not only a new form of communication, but also a powerful tool that can be used to gather information related to relevant questions, for example: which is the favorite political party for the upcoming elections, what are the most talked about movies of the past year, which is the best rated restaurant in a given area, etc. Extracting relevant information from social networks is a topic of interest, mainly due to the enormous amount of potential data available. However, traditional network analysis techniques are becoming obsolete due to the exponential growth of social networks, in terms of the number of active users. Social network analysis has become one of the most popular and challenging tasks in data science [2]. One of the most addressed problems in social networks is the analysis of the relevance of users in a given social network [3]. The relevance of a user is commonly related to the number of followers or friends the user has in a given social network. However, this concept can be extended, as a user can be relevant not only if he/she is connected to a large number of users, but also if he/she is connected to relevant users. Several metrics have been proposed to analyze the relevance of a user in a social network, with PageRank emerging as one of the most widely used [4]. Moreover, it is interesting to know which users will be the most relevant before they become influential [5]. Finally, in the field of marketing analytics, there is a special interest in generating the profile of a user given a set of tweets written by that user [6]. Assessing the relevance of a user has become a more complex problem that consists of detecting specific users (often referred to as influencers) with certain attributes that may be personal (credibility or enthusiasm) or related to their social networks (connectedness or centrality). These attributes allow them to influence a large number of users, either directly or indirectly [7]. Another important issue related to the influence of people on other users is sentiment analysis in social networks. This analysis focuses on discovering what people think about a given topic by analyzing the information they post on social networks. A complete survey on sentiment analysis techniques can be found in [8]. The problems described above deal only with individual users. However, there are also some problems related to the network structure, dedicated to finding specific attributes and properties that can help to infer additional information from

This research was funded by "Ministerio de Ciencia, Innovación y Universidades" under grant ref. PGC2018-095322-B-C22, "Comunidad de Madrid" and "Fondos Estructurales" of European Union with grant refs. S2018/TCS-4566, Y2018/EMT-5062.

the whole social network. In this context, community detection emerges as one of the most studied problems. Most social networks present a common characteristic called community structure [9]. Networks with this property can be divided into groups in such a way that connections between users in the same group are dense, while connections between users in different groups are sparse. The connections between users can represent different characteristics depending on the social network and the user's profile, i.e., from professional relationships to friendships or common hobbies. Community detection tasks are dedicated to find and analyze these groups to better understand and visualize the structure of the network and the relationships between its users. Running community detection algorithms in today's social networks is computationally intensive, mainly due to their continuous growth. Moreover, since these networks are constantly changing (new friendships, mentions to users, viral information, etc.), it is interesting to perform community detection in the shortest possible computational time, producing information in real time. These characteristics make traditional exact methods unsuitable for the current size of social networks, requiring heuristic algorithms to speed up the process without losing quality. Recent works have addressed the community detection problem from a non-exact perspective to generate high quality solutions in small computational time [10]. Several studies are devoted to reduce the computational effort to detect communities in social networks [11], [12]. When comparing traditional algorithms in large modern social networks, it can be observed that some of the algorithms require more than 40000 seconds for networks with approximately 10000 nodes, and cannot provide a solution after 24 hours of computation for networks with more than 50000 nodes. The growth of social networks complicates their representation and understanding. Communities in a social network generally summarize the entire network, but reducing its size and thus making it easier to analyze. In addition, community detection in social networks has several practical applications. Recommendation systems leverage the data of similar users to suggest content that may be of interest to them. To find similar users in a network, we can simply perform community detection across the network [13] improving the results of the recommendation system. Communities in social networks also identify people with similar interests, allowing us to evaluate the popularity of a political party [14], or even detect radicalism in social networks [15]. While there are several community detection algorithms that have been proposed with the objective of identifying similar users in networks, most of the available algorithms have been designed to optimize a specific objective function, making it difficult to adapt the algorithm to a different one. However, the continuous evolution of this area results in a continuous proposal of new metrics that better represent the community structure of a given network. In this thesis, efficient and versatile algorithms will be developed that can be easily adapted to different optimization metrics.

II. Hypotheses and objectives

Once the problem to be solved has been identified, the next point to be addressed during the development of a research project is the formulation of an initial hypothesis. This hypothesis is a tentative proposal that seeks to formulate a solution to the problem posed. The hypothesis will represent a fundamental element in the research process, since it will serve as a guide.

The hypothesis proposed for the development of this Doctoral Thesis can be summarized in the following terms: the problem of community detection in social networks is a problem with a high computational cost, with practical interest in different scientific disciplines. Therefore, it is interesting to develop algorithms that are able to solve it efficiently, obtaining optimal solutions if possible or, at least, of high quality, in a reasonable amount of time. For this reason, heuristic algorithms become relevant to solve this type of problems. In recent years, bioinspired heuristics have been proposed to avoid the computational requirements of exact implementations. In the area of bioinspired computation, evolutionary approaches are the most popular. It is important to highlight the review performed by Pizzuti in [16] about Evolutionary Computation (EC) techniques to detect communities in networks. An interesting work about EC is the work published by Said et. al [17], where authors designed a clustering coefficient-based genetic algorithm able to detect cohesive groups from dense graphs and also, communities in sparse networks. Other relevant work is [18] that presents a genetic algorithm that uses a multi-individual ensemble learning-based crossover function. The algorithm is improved with a local search strategy to speed up the convergence.

Other well-known bioinspired algorithms are the ones belonging to swarm intelligence. In this new group, the most popular algorithms are Particle Swarm Optimization (PSO) and Ant Colony Optimization (ACO). These two algorithms are inspired by the social behaviour of birds within a flock, and the behaviour of ants seeking a path from the nest to the source of food, respectively. PSO has been successfully used for CDP in [19], where a discrete PSO algorithm is used to extract the communities in large-scale social networks by optimizing the modularity. Regarding ACO algorithm, this algorithm has been used to extract high-quality communities in Ego Networks [10].

However, this family of algorithms has the disadvantage that they do not take advantage of specific information from the network. In this Thesis, the development of algorithms based on traditional metaheuristics, performing an in-deep study of the networks under evaluation opens a new perspective for future research for community detection problems.

The proposed heuristic algorithms will make use of metaheuristic techniques, which have been shown to be effective procedures when dealing with optimization problems. In particular, population-based metaheuristics constitute a subset of this type of techniques, characterized by taking into consideration more than one solution simultaneously and providing

combination mechanisms among them. Techniques such as Scatter Search are a clear example of this type of metaheuristics. On the other hand, the so-called trajectory-based metaheuristics start from an initial solution and are capable of generating a trajectory in the solution space. Examples of this type of metaheuristics are GRASP (Greedy Randomized Adaptive Search Procedure) or Path Relinking. The main objective of the Doctoral Thesis is to develop algorithms that solve several optimization problems related to the analysis of social networks and the detection of communities in them, approaching them from a heuristic point of view, both from the mono-objective and multi-objective perspective.

A. Methodology used to reach the main objective

The proposed heuristic algorithm will be complemented with metaheuristic techniques best suited to the problem. In particular, GRASP, Variable Neighborhood Search or Path Relinking will be considered. To achieve the main objective described, it is necessary to cover the following partial objectives:

- To know the state of the art of the problem.
- To design and develop a heuristic algorithm for solving the problem.
- To validate the heuristic algorithm.
- To experimentally compare the proposed algorithm with the state of the art algorithms.
- Submit the partial results of the research work to review processes by independent institutions.
- To write a research report.

For the correct development of the objectives of this Doctoral Thesis, the necessary instrumentation is based on a high performance computer that allows the development of the proposed algorithms, as well as to carry out the computational experiments necessary to evaluate the quality of these algorithms. It would be advisable to have more than one computer to be able to carry out experiments in parallel in time, which would accelerate the obtaining of results, as well as broaden the spectrum of problems that we can face. The procedure to be followed to achieve the proposed objectives is based on the scientific method. First, the optimization problem to be solved is identified. During this stage, it is verified that the problem is of interest and that, in addition, it leads to practical applications so that there can be a transfer of knowledge. Once the problem has been identified, the process begins with a review of the state of the art. As a result of this phase, the algorithms previously developed for the problem in question are identified, in addition to the set of instances used in their evaluation. According to the study carried out, a hypothesis for the resolution of the problem is then proposed, which will lead to the development of one or more specific algorithms. The validation of these algorithms will be carried out in two stages: first, the developed code is checked for errors and, then, the strategies used are validated.

If the result of this validation is positive, a final evaluation of the proposal is made, comparing the results obtained with those of other algorithms present in the state of the art. In this comparison, it will be necessary to use statistical tests to confirm that the proposed algorithm is a relevant scientific contribution to the area.

B. Milestones that mark the fulfillment of the proposed objective

The main objective of this Thesis is considered fulfilled if the developed algorithms and their respective associated work derive in high impact publications in scientific journals. Furthermore, if these works derive in a relevant transference to the society, this will demonstrate the importance and impact of the Thesis, reflecting the need for the development of the previously exposed algorithms in the field of community detection in a real environment.

C. Milestones achieved to date

In this section the milestones reached to date are exposed. Conference and journal publications are listed and, subsequently, problem definition, algorithmic proposal and obtained result and conclusions are briefly explained.

To date, the milestones achieved are as follows:

- Pérez-Peló, S., Sánchez-Oro, J., and Duarte, A. Detecting Weak Points in Networks Using Variable Neighborhood Search. *ICVNS 2021, Lecture Notes in Computer Science*, vol. 11328, pp.141-151, Springer, Cham (2018) [20].
  - Index: SJR
  - Impact Factor: 0.28
  - Rank: Q2
- Pérez-Peló, S., Sánchez-Oro, J., and Duarte, A. A Metaheuristic Approach for the α-separator Problem. *IDEAL 2018, Lecture Notes in Computer Science*, vol. 11328, pp. 336-343, Springer, Cham (2018) [21].
  - Index: SJR
  - Impact Factor: 0.28
  - Rank: Q2
- Pérez-Peló, S., Sánchez-Oro, J., Martín-Santamaría, R., and Duarte, A. On the Analysis of the Influence of the Evaluation Metric in Community Detection over Social Networks. *Electronics*, 8(1), 23 (2019) [22].
  - Index: JCR
  - Impact Factor: 1.764
  - Rank: 154 / 266, Q2
- Pérez-Peló, S., Sánchez-Oro, J., and Duarte, A. Finding weaknesses in networks using greedy randomized adaptive search procedure and path relinking. *Expert Systems*, 37(6), e12540 (2020) [23].
  - Index: JCR
  - Impact Factor: 1.505
  - Rank: 49 / 105, Q2
- Pérez-Peló, S., Sánchez-Oro, J., López-Sánchez, A. D., and Duarte, A. A Multi-Objective Parallel Iterated Greedy for Solving the p-Center and p-Dispersion Problem. *Electronics*, 8(12), 1440 (2019).
  - Index: JCR
  - Impact Factor: 1.764
  - Rank: 154 / 266, Q3

1) Detecting weak points in networks: In order to make an initial approach to the Thesis topic, it is interesting to analyze how a network could be divided in several communities/groups by removing certain elements from them. Concretely, if we look for those elements whose removal implies the network disarticulation, we can obtain a first idea about how the actors in this network are related among them. It is an idea related to the detection of articulation points in graphs: by detecting the points whose elimination leads to the splitting of the network, we can get an idea of how to split the network into cohesive groups.

In this work [23], obtained results show that a heuristic approach is capable to overcome the state of the art in terms of finding critical nodes in networks, whose removal derives in the network division in different groups denoted by the resulting connected components in the graph. It is worth mentioning that this work has been presented in a conference paper also [20], demonstrating the impact and relevance of the proposed method.

2) On the Analysis of the Influence of the Evaluation Metric in Community Detection over Social Networks: Once the first approach revealed that networks have a certain inherent community structure, the community detection problem is addressed as an optimization problem, where the modularity maximization is the objective function. In order to solve this problem, in [22] a GRASP based algorithm is proposed, obtaining high-quality solutions in low computing times. In this paper, the superiority of the GRASP approach with respect to a set of previous algorithms in literature is exposed, providing high quality results in terms of modularity and conductance metrics. The conductance metric is used to evaluate the quality of all algorithms under evaluation, given that it is an objective metric that has not been used as objective function to solve the problem. The proposed GRASP algorithm, combined with a local search strategy, overcomes the state-of-the-art methods.

3) Multi-objective community detection problem: The research in the single objective version of the community detection problem lead us to know that there are a lot of metrics that can be used as objective function to solve it, so the idea of approaching the problem as a multi-objective problem emerged. In a multi-objective context, two or more conflicting objectives are optimized simultaneously. These objective are considered in conflict if it is not possible to improve one of them without making the other worse. In this context, the multi-objective community detection problem has aroused as a bi-objective problem where ratio cut and negative ratio association are tried to be minimized. These objectives are focused in maximize the connections between nodes belonging to the same community and minimize those who connect nodes in different communities. Although the objective is shared, the way in which these metrics are evaluated means that optimizing each of them separately leads to completely different solutions. These are two objectives that have proven to be in conflict [24]. To tackle this problem, a Multi-objective Variable Neighborhood Search variant is proposed. This approach modifies the original VNS algorithm in such a way that a solution is conformed by an entire non-dominated set of solutions, instead of a single solution. Using this modification, we can address the problem from a multi-objective point of view. Another important modification is that an improvement is considered if the non-dominated front is changed in an iteration of the VNS. It is because a modification in the reference front implies that a new non-dominated solution has entered into it, while one or more dominated solutions has been removed, improving the quality of the front under construction. Evaluating the obtained results using a set of well-known multi-objective metrics, such as coverage, hypervolume or ϵ-indicator, comparing the algorithm obtained front with the reference front built with the best solutions found by all algorithms under evaluation, the proposed methodology shows high performance and quality. Furthermore, evaluating the best solutions found in the front in terms of normalized mutual information (NMI) and modularity, this proposal obtains high-quality results in less computing time than the best previous algorithm found in literature. These results prove that a multi-objective approach can be applied in order to solve community detection problems, obtaining competitive results in reasonable computation times. This work has been sent to Applied Soft Computing journal, ranked in Q1 of JCR, with an index factor of 5.472. Actually it is in second review and will be published shortly.

4) Overlapping community detection: Given that community detection problem can be addressed from both a single-objective and a multi-objective perspective, the next step is to check if a more realistic community detection can be performed. In real world social networks, a user typically belongs to more than one community at the same time. For example, a user interested in informative scientific papers could be interested in football or other sports simultaneously. In this context, the overlapping community detection problem emerges as an interesting problem to be solved using an heuristic point of view. In this work, a metaheuristic that combines GRASP and Iterated Greedy is presented. In order to taking into account the possibility of overlapped communities, the metric used to evaluate the results should be modified. An overlapping based modularity modification is taken into account in order to evaluate the quality of the solutions reached by the proposed algorithm. The performed experiments show that the proposal is able to reach high quality solutions in networks with an elevated number of nodes and edges in reasonable computing times. The proposal overcomes the best algorithm found in literature, both in terms of computational time and quality. It is demonstrated that a more realistic model can be solved by the metaheuristic point of view, allowing a more interesting transference from the scientific community to the society, given that this approach can be applied in a wide amount of real-life scenarios.

5) Solving the multi-objective p-Center, p-Dispersion problem: Having obtained a solid knowledge about the multi-objective approach, an opportunity to collaborate in another bi-objective problem with external researchers emerged. This work is not exactly related to community detection, but it is

also applicable to social networks context, so the collaboration is justified. In this work, the classical p-Center and p-Dispersion problems are solved simultaneously. In the context of social networks, it can be applied to find users within a network that can be selected to spread some advertisement to reach the more users as possible among their followers. A metaheuristic based in Iterated Greedy is proposed, adapting the framework to the multi-objective context, in order to provide an algorithm that fits better in real-world applications. In this context, a solution is defined by all non-dominated solutions that conform the non-dominated front. This algorithm is able to find a high quality approximation front in short computing times, emerging as the most competitive algorithm from a heuristic point of view in the state of the art.

D. Current research lines

So far, all the research carried out has been based on the idea that the networks studied have a specific structure that does not change over time. However, the reality is that social networks are constantly changing and growing, so it would be interesting to study the application of metaheuristic algorithms in a field in which the networks on which these algorithms are applied are constantly changing. In this context, the dynamic social networks analysis takes relevance for the present doctoral thesis. The work is currently focused in making a literature review of this problem set. Once this process is completed, a new approach based in metaheuristic algorithms will be proposed, trying to provide new competitive procedures that allow reach high quality solutions in lower computing times.

III. Conclusions and future work

Social network analysis and, more specifically, community detection problems, are an actual and interesting research area. The real-life applications of this kind of problems are extensive, and having fast and good algorithms in terms of quality of solutions is an important task, both for the scientific community and for companies interested in applying beneficial solutions in this field. It is a very broad field, with many branches open for exploration and study. Therefore, it is important to generate knowledge and competitive algorithms for those situations in which obtaining a low computation time is a main objective, especially in such a changing environment. In this context, heuristic and metaheuristic algorithms offer an efficient alternative to classical algorithms. This doctoral thesis focuses on providing new algorithms that make a high contribution to the scientific community in this field, as well as to address community detection problems that have not been addressed from the metaheuristics perspective.

As a future work, the aforementioned dynamic social networks analysis is an important and interesting branch. If the results obtained by the current research shows that the metaheuristic approach has relevance, new lines can be proposed in the context of dynamic social network analysis, as multi-objective proposals or the existence of overlapping communities.

Acknowledgment

This research was funded by “Ministerio de Ciencia, Innovación y Universidades” under grant ref. PGC2018-095322-B-C22, “Comunidad de Madrid” and “Fondos Estructurales” of European Union with grant refs. S2018/TCS-4566, Y2018/EMT-5062.

References

[1] S. P. Borgatti, M. G. Everett, and J. C. Johnson, *Analyzing social networks*. Sage, 2018.

[2] S. N. Dorogovtsev and J. F. Mendes, *Evolution of networks: From biological nets to the Internet and WWW*. OUP Oxford, 2013.

[3] J. Tang, J. Sun, C. Wang, and Z. Yang, “Social influence analysis in large-scale networks,” in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2009, pp. 807–816.

[4] L. Page, S. Brin, R. Motwani, and T. Winograd, “The pagerank citation ranking: Bringing order to the web.” Stanford InfoLab, Tech. Rep., 1999.

[5] K. Almgren and J. Lee, “An empirical comparison of influence measurements for social network analysis,” *Social Network Analysis and Mining*, vol. 6, no. 1, pp. 1–18, 2016.

[6] K. Ikeda, G. Hattori, C. Ono, H. Asoh, and T. Higashino, “Twitter user profiling based on text and community mining for market analysis,” *Knowledge-Based Systems*, vol. 51, pp. 35–47, 2013.

[7] M. GLADWELL and T. Point, “How little things can make a big difference,” 2000.

[8] R. K. Bakshi, N. Kaur, R. Kaur, and G. Kaur, “Opinion mining and sentiment analysis,” in *2016 3rd international conference on computing for sustainable global development (INDIACom)*. IEEE, 2016, pp. 452–455.

[9] M. Girvan and M. E. Newman, “Community structure in social and biological networks,” *Proceedings of the national academy of sciences*, vol. 99, no. 12, pp. 7821–7826, 2002.

[10] A. Gonzalez-Pardo, J. J. Jung, and D. Camacho, “Aco-based clustering for ego network analysis,” *Future Generation Computer Systems*, vol. 66, pp. 160–170, 2017.

[11] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, “Fast unfolding of communities in large networks,” *Journal of statistical mechanics: theory and experiment*, vol. 2008, no. 10, p. P10008, 2008.

[12] P. Pons and M. Latapy, “Computing communities in large networks using random walks,” in *International symposium on computer and information sciences*. Springer, 2005, pp. 284–293.

[13] C. Cao, Q. Ni, and Y. Zhai, “An improved collaborative filtering recommendation algorithm based on community detection in social networks,” in *Proceedings of the 2015 annual conference on genetic and evolutionary computation*, 2015, pp. 1–8.

[14] N. Zalmout and M. Ghanem, “Multidimensional community detection in twitter,” in *8th International Conference for Internet Technology and Secured Transactions (ICITST-2013)*. IEEE, 2013, pp. 83–88.

[15] D. Camacho, I. Gilpérez-López, A. Gonzalez-Pardo, A. Ortigosa, and C. Urruela, “Risktrack: a new approach for risk assessment of radicalisation based on social media data,” in *CEUR Workshop Proceedings*, 2016.

[16] C. Pizzuti, “Evolutionary computation for community detection in networks: A review,” *IEEE Transactions on Evolutionary Computation*, vol. 22, no. 3, pp. 464–483, 2018.

[17] A. Said, R. A. Abbasi, O. Maqbool, A. Daud, and N. R. Aljohani, “CC-GA: A clustering coefficient based genetic algorithm for detecting communities in social networks,” *Applied Soft Computing*, vol. 63, pp. 59–70, 2018.

[18] M. Moradi and S. Parsa, “An evolutionary method for community detection using a novel local search strategy,” *Physica A: Statistical Mechanics and its Applications*, vol. 523, pp. 457–475, 2019.

[19] Q. Cai, M. Gong, L. Ma, S. Ruan, F. Yuan, and L. Jiao, “Greedy discrete particle swarm optimization for large-scale social network clustering,” *Information Sciences*, vol. 316, pp. 503–516, 2015.

[20] S. Pérez-Peló, J. Sánchez-Oro, and A. Duarte, “Detecting weak points in networks using variable neighborhood search,” in *International Conference on Variable Neighborhood Search*. Springer, 2018, pp. 141–151.

[21] S. Pérez-Peló, J. Sánchez-Oro, and A. Duarte, "A metaheuristic approach for the α-separator problem," in *International Conference on Intelligent Data Engineering and Automated Learning*. Springer, 2018, pp. 336–343.

[22] S. Pérez-Peló, J. Sanchez-Oro, R. Martin-Santamaria, and A. Duarte, "On the analysis of the influence of the evaluation metric in community detection over social networks," *Electronics*, vol. 8, no. 1, p. 23, 2019.

[23] S. Pérez-Peló, J. Sánchez-Oro, and A. Duarte, "Finding weaknesses in networks using greedy randomized adaptive search procedure and path relinking," *Expert Systems*, vol. 37, no. 6, p. e12540, 2020.

[24] F. Cheng, T. Cui, Y. Su, Y. Niu, and X. Zhang, "A local information based multi-objective evolutionary algorithm for community detection in complex networks," *Applied Soft Computing*, vol. 69, pp. 357 – 367, 2018.

Optimización heurística de problemas relacionados con redes sociales

Isaac Lozano Osorio
Universidad Rey Juan Carlos
Móstoles, Madrid
isaac.lozano@urjc.es

Datos de la Tesis Doctoral

- Datos de contacto: isaac.lozano@urjc.es, jesus.sanchezoro@urjc.es, abraham.duarte@urjc.es.
- Título para la tesis: Optimización heurística de problemas relacionados con redes sociales.
- Director(es): Jesús Sánchez-Oro Calvo y Abraham Duarte Muñoz.
- Departamento: Ciencias de la Computación, Arquitectura de Computadores, Lenguajes y Sistemas Informáticos y Estadística e Investigación Operativa.
- Fecha de inicio: Noviembre de 2020.

***Resumen*—El crecimiento de las redes sociales y la gran cantidad de datos que generan han despertado un gran interés en el ámbito científico y empresarial. Debido a esto, han surgido diferentes problemas que son difíciles de resolver con técnicas clásicas, según se ha demostrado en el estado del arte. Entre estos problemas se puede destacar el marketing viral, el análisis de la expansión de enfermedades o el análisis de la influencia de los usuarios, entre otros. El estudio y resolución de problemas complejos mediante el uso de metaheurísticas es un tema muy extendido en el área de la inteligencia artificial, pero no ha sido estudiado en profundidad en este tipo de problemas. El objetivo principal de esta Tesis Doctoral es la aplicación de técnicas heurísticas y metaheurísticas en problemas sobre redes sociales para proporcionar soluciones competitivas con el estado del arte.**

***Palabras clave*—Redes sociales, maximización de la influencia, teoría de grafos, marketing viral, GRASP.**

I. Introducción

En el ámbito informático, algunos tipos de problemas de optimización son relativamente fáciles de resolver, ya que sus características permiten el uso de técnicas deterministas capaces de encontrar su solución óptima. En este proyecto de Tesis Doctoral se abordan problemas reales que no pueden ser resueltos con algoritmos exactos, bien porque sus características no han permitido el desarrollo de ninguna técnica exacta que asegure la localización de la solución óptima, o porque aun pudiendo ser las técnicas exactas utilizadas, el tiempo necesario para obtener la solución del problema resulte prohibitivo. Por ello, este tipo de problemas se suele resolver mediante el uso de técnicas heurísticas, las cuales normalmente permiten generar una solución factible (que cumple las restricciones del problema) que es evaluada utilizando una función objetivo determinada. Estos problemas se pueden aplicar a diferentes ámbitos como salud, industria, transporte, etc.

La evolución y la expansión de las redes sociales han comenzado a atraer el interés de la comunidad científica en los últimos años. Debido a esto, han surgido numerosos problemas que son difíciles de resolver con técnicas clásicas, como pueden ser el marketing viral, el análisis de la expansión de enfermedades y el análisis de la influencia de los usuarios, entre otros [2, 6, 7, 10, 9]. Tanto el ámbito empresarial como el científico tiene interés en resolver problemas de esta familia, ya que tienen aplicaciones en ámbitos tan diversos como maximizar el beneficio de la promoción de un producto o minimizar el contagio de una pandemia. Esta familia de problemas se suele conocer como problemas de maximización de la influencia en redes sociales (SNIM, del inglés *Social Network Influence Maximización*), y tienen como objetivo encontrar a los usuarios más influyentes dentro de una red social, mediante la simulación de un modelo de difusión de la influencia [5]. Los problemas de la familia de SNIM son problemas de la familia $\mathcal{NP}$-difícil y, además, la mayoría de las redes sociales mantienen a un elevado número de usuarios activos, por lo que se han convertido en un reto real para la comunidad científica. En varios trabajos se han propuesto aproximaciones basadas en heurísticas y metaheurísticas para resolver este tipo de problemas, como por ejemplo algoritmos evolutivos, pero no pueden enfrentarse a redes reales debido a limitaciones de cómputo. Esto se debe a que la evaluación de una solución al problema requiere de un esfuerzo computacional muy elevado. La aplicación de metaheurísticas eficientes en el ámbito de las redes sociales permitirá disponer de herramientas que permitan analizar la influencia de los usuarios en tiempo real, lo que supone un avance muy significativo en este ámbito.

Una vez identificado el problema que se pretende resolver, el desarrollo de una investigación prosigue con la formulación de una hipótesis inicial. Dicha hipótesis es una propuesta tentativa que pretende formular una solución al problema abordado. La hipótesis que se plantea para el desarrollo de esta Tesis Doctoral se puede resumir en los siguientes términos: el problema de la maximización de la influencia en redes sociales (SNIM) es un problema $\mathcal{NP}$-difícil, con interés práctico en numerosas disciplinas científicas. Por ello, es interesante el desarrollo de algoritmos que sean capaces de generar soluciones de alta calidad en un tiempo razonable. Para la resolución de este tipo de problemas son relevantes

los algoritmos de resolución heurística, que son capaces de obtener soluciones de alta calidad en cortos períodos de tiempo. Se propone además el uso de técnicas metaheurísticas, las cuales se han mostrado como procedimientos eficaces a la hora de abordar problemas de optimización. En particular, las metaheurísticas poblacionales constituyen una subfamilia de este tipo de técnicas, caracterizada por considerar más de una solución de manera simultánea y proporcionar mecanismos de combinación entre ellas. Búsqueda Dispersa es un claro exponente de este tipo de metaheurísticas. Por otro lado, las denominadas metaheurísticas trayectoriales parten de una solución inicial y son capaces de generar una trayectoria en el espacio de soluciones. GRASP (del inglés *Greedy Randomized Adaptive Search Procedure*) es un ejemplo de metaheurística trayectorial. El algoritmo heurístico propuesto se complementará con las técnicas metaheurísticas que mejor se adecúen al problema, en particular se considerarán GRASP y Búsqueda Dispersa.

II. Metodología y Plan de Trabajo

La metodología usada en esta Tesis Doctoral se basará en el método científico [4], y busca validar nuevos conocimientos siguiendo unos determinados pasos: observación, hipótesis, experimentación, medición, validación, reproducibilidad, revisión por pares y publicación. El método científico es muy utilizado y se aplica a muchas áreas. En el ámbito de la optimización heurística y en esta Tesis Doctoral será utilizado como guía para proponer soluciones basadas algoritmos heurísticos y algoritmos metaheurísticos como se muestra en la Figura 1. Determinar si una propuesta algorítmica es mejor que el estado

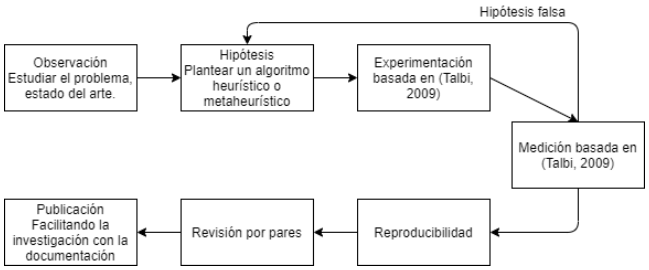


Figura 1. Método científico aplicado a la optimización algorítmica

del arte debe hacerse de una manera justa. Para evaluar un procedimiento metaheurístico desde un punto de vista objetivo, en Talbi, E. G. [11] se proponen tres pasos: diseño de la experimentación, ejecución y documentación de los resultados. El diseño experimental busca definir los objetivos de los experimentos mostrando las instancias (diferentes situaciones para cada uno de los problemas abordados) que se usarán para validar el algoritmo propuesto. Es recomendable usar las mismas instancias propuestas en los trabajos anteriores del estado del arte. De esa manera, la comparación entre el algoritmo propuesto y los métodos previos es más objetiva y justa. Una vez diseñado el experimento, debemos conocer los indicadores de rendimiento del algoritmo. Para validar los resultados, es necesario llevar a cabo un análisis estadístico de ellos. En los métodos de optimización exacta, la eficiencia en términos de tiempo de ejecución es el indicador principal. Sin embargo, para evaluar el rendimiento de los procedimientos heurísticos y metaheurísticos deben considerarse otros indicadores: calidad de la solución, esfuerzo computacional y robustez del algoritmo [1, 3]. Finalmente, la interpretación de los resultados obtenidos, además de mostrarlos, realizar un análisis siguiendo los objetivos definidos, para poder llevar a cabo una comparación justa. Para ello, es importante que los experimentos se ejecuten en un mismo entorno informático de experimentación. La interpretación de los resultados debe ser explícita, basarse en los objetivos definidos y las medidas de rendimiento consideradas. Para cumplir el método científico en este ámbito la reproducibilidad de los experimentos es algo importante. Para ello, será necesario documentar todas las decisiones tomadas en los dos primeros pasos y facilitar a quien lo requiera la investigación publicada.

II-1. Objetivos: Como se ha mencionado anteriormente, el objetivo principal es la aplicación de técnicas heurísticas y metaheurísticas en problemas sobre redes sociales para proporcionar soluciones competitivas con el estado del arte. Para alcanzar este objetivo principal, es necesario cubrir los siguientes objetivos parciales:

- Conocer el estado del arte del problema.
- Diseñar y desarrollar un algoritmo heurístico para la resolución del problema SNIM utilizando el modelo de difusión ICM.
- Validar el algoritmo heurístico.
- Comparar experimentalmente el algoritmo propuesto, con los algoritmos del estado del arte.
- Desarrollar un nuevo algoritmo que sea capaz de maximizar la influencia utilizando el modelo WCM.
- Validar el nuevo algoritmo propuesto.
- Analizar los resultados obtenidos por el nuevo algoritmo y compararlos con los algoritmos del estado del arte.
- Adaptar los algoritmos propuestos al problema de la contención de la evolución de pandemias.
- Validar los resultados de la adaptación.
- Someter los resultados parciales del trabajo de investigación a procesos de revisión por parte de instituciones independientes.
- Redactar una memoria de investigación.

II-2. Plan de Trabajo: La planificación se ha desglosado en los cuatro años de duración de la siguiente manera:

- Primer año: Propuesta de un algoritmo heurístico para el SNIM con el modelo de difusión ICM.
- Segundo año: Propuesta de un algoritmo heurístico para el SNIM con el modelo de difusión WCM.
- Tercer año: Adaptación de los algoritmos diseñados para la aplicación directa sobre la contención de la expansión de pandemias.
- Cuarto año: Escritura de la memoria de investigación.

III. Relevancia

Esta Tesis Doctoral, gracias a la diversidad de ámbitos dónde se puede aplicar, tiene una relevancia notable tanto en

el ámbito de la investigación como en el de la transferencia a la sociedad.

En el ámbito académico, el algoritmo propuesto será de gran utilidad para la comunidad científica, ya que en los problemas relacionados con redes sociales, las propuestas basadas en heurísticas y metaheurísticas no han sido estudiadas con mucho detalle. La aplicación de metaheurísticas eficientes en el ámbito de las redes sociales permitirá disponer de herramientas que permitan analizar la influencia de los usuarios en tiempo real, lo que supone un avance muy significativo en este ámbito. Además permitirá proporcionar un marco base de comparación para cualquier nueva propuesta algorítmica.

En cuanto a la transferencia a la sociedad dentro de las diferentes aplicaciones, las empresas de publicidad que realicen marketing de *influencers* (campañas en las que se seleccionan diferentes personas para promocionar un producto o servicio), podrán conocer qué personas son las indicadas para realizar la campaña según los diferentes objetivos que se propongan (ventas, visibilidad, etc). En la actualidad uno de los problemas presentes es la erradicación de la pandemia mundial a la que nos enfrentamos. En este contexto, el algoritmo permite determinar que nodos se deben aislar para reducir drásticamente el número de contagios.

Los resultados parciales obtenidos por la Tesis Doctoral en desarrollo durante el primer año son los siguientes. Participación en 2 congresos internacionales (con revisión por pares), ICVNS (http://www.icvns2020.info/) y PPSN (https://ppsn2020.liacs.leidenuniv.nl/), se han enviado 2 trabajos en revistas indexadas en el *Journal of Citation Reports* (JCR) a la espera de revisión y se ha publicado 1 trabajo en una revista JCR indexada en el segundo cuartil [8].

Agradecimientos

El doctorado está siendo financiado por el Ministerio de Ciencia, Innovación y Universidades con la subvención ref. PGC2018-095322-B-C22, Çomunidad de Madridz ”Fondos Estructurales”de la Unión Europea con las referencias de subvención S2018/TCS-4566, Y2018/EMT-5062, PEJD-2019-PRE/TIC-16151. Por el Programa de Atracción de Talento Investigador de la Comunidad de Madrid con la referencia de subvención 2017-T2/TIC-5664 y por la ayuda de la Universidad Rey Juan Carlos 2021/PREDOC21-007-2145.

Bibliografía

[1] Richard S. Barr et al. «Designing and reporting on computational experiments with heuristic methods». En: *Journal of Heuristics* 1.1 (sep. de 1995), págs. 9-32. DOI: 10.1007/bf02430363.

[2] Adam D’angelo et al. *Targeting Advertisements in a Social Network*. US Patent App. 12/853,242. Dic. de 2010.

[3] Bruce L. Golden y Arjang A. Assad. «A decision-theoretic framework for comparing heuristics». En: *European Journal of Operational Research* 18.2 (nov. de 1984), págs. 167-171. DOI: 10.1016/0377-2217(84)90182-6.

[4] Hugh G. Gauch Jr. *Scientific Method in Practice*. Cambridge University Press, dic. de 2002. DOI: 10.1017/cbo9780511815034.

[5] David Kempe, Jon Kleinberg y Éva Tardos. «Maximizing the spread of influence through a social network». En: *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining - KDD ’03*. ACM Press, 2003. DOI: 10.1145/956750.956769.

[6] Stephen F. King y Thomas F. Burgess. «Understanding success and failure in customer relationship management». En: *Industrial Marketing Management* 37.4 (jun. de 2008), págs. 421-431. DOI: 10.1016/j.indmarman.2007.02.005.

[7] Alden S. Klovdahl. «Social networks and the spread of infectious diseases: The AIDS example». En: *Social Science & Medicine* 21.11 (ene. de 1985), págs. 1203-1216. DOI: 10.1016/0277-9536(85)90269-2.

[8] Isaac Lozano-Osorio et al. «Optimizing Computer Networks Communication with the Band Collocation Problem: A Variable Neighborhood Search Approach». En: *Electronics* 9.11 (2020), pág. 1860.

[9] Sergio Pérez-Peló, Jesús Sánchez-Oro y Abraham Duarte. «Finding weaknesses in networks using Greedy Randomized Adaptive Search Procedure and Path Relinking». En: *Expert Systems* 37.6 (feb. de 2020). DOI: 10.1111/exsy.12540.

[10] Sergio Pérez-Peló et al. «On the Analysis of the Influence of the Evaluation Metric in Community Detection over Social Networks». En: *Electronics* 8.1 (dic. de 2018), pág. 23. DOI: 10.3390/electronics8010023.

[11] El-Ghazali Talbi. *Metaheuristics: From Design To Implementation*. Wiley Publishing, 2009. ISBN: 0470278587.

Mathematical Modeling and Analysis of Combinatorial Optimization Problems

Imanol Unanue
Intelligent Systems Group (ISG)
Department of Computer Science and Artificial Intelligence
University of the Basque Country UPV/EHU
Donostia-San Sebastián, Spain
imanol.unanue@ehu.eus

María Merino (Advisor)
Basque Center for Applied Mathematics (BCAM), Bilbao, Spain
Department of Mathematics
University of the Basque Country UPV/EHU
Leioa, Spain
maria.merino@ehu.eus

Jose A. Lozano (Advisor)
Basque Center for Applied Mathematics (BCAM), Bilbao, Spain
Intelligent Systems Group (ISG)
Department of Computer Science and Artificial Intelligence
University of the Basque Country UPV/EHU
Donostia-San Sebastián, Spain
ja.lozano@ehu.eus

***Abstract*—Combinatorial Optimization Problems have always been engaging among researchers. Due to the inefficiency of solving these problems in an exhaustive way and the uncertainty of the quality of the obtained solutions by the metaheuristic algorithms, the development of the theoretical study of these problems has gained interest. In this thesis, Combinatorial Optimization Problems are analyzed by means of metaheuristic algorithms. This thesis started with the theoretical study of Estimation of Distribution Algorithms based on distance-based exponential models, following similar motivations to the works presented in our research group *Intelligent Systems Group*. Based on the results obtained, the thesis refocused its main objective and in a second phase of this thesis, we try to generate a taxonomy of binary-based Combinatorial Optimization Problems, with the final goal of finding a matching between problem instances and metaheuristic algorithms. The presented mathematical framework to achieve our target is based on Walsh decomposition.**

***Index Terms*—Combinatorial Optimization Problems, Estimation of Distribution Algorithms, Mallows Model, Walsh Functions.**

I. Introduction

Combinatorial Optimization Problems (COPs) are a class of problems in which the objective is to find the best solution from a discrete search space according to one or more fitness functions. When the solutions have several constraints to fulfill, then the COP is called constrained; otherwise, the COP is unconstrained. Overall, COPs are defined as maximization (minimization) problems [1], [2].

COPs appear in many fields of science and engineering, among others [3]. However, the main disadvantage of the majority of this kind of problems is that frequently it is not efficient to calculate all the fitness function values and compare all the solutions to ensure which the best solution is. This difficulty is explained as the most important problem in computer science and is related to one of the Millennium Prize Problems: *P* vs *NP* Problem [4].

In order to face this difficulty, researchers have developed new strategies and algorithms to obtain the optimal solution of a problem without calculating all the possible solutions. One of the most popular technique to solve COPs is the development of metaheuristic algorithms.

In the last 30 years, metaheuristic algorithms have become in an efficient alternative to the classical methods to solve COPs. Due to the extensive range of algorithms used in the literature to solve the problems, several classifications of the algorithms have been proposed. The most common classification focuses on how the algorithm generates new solutions at each iteration of the algorithm, and there are three kind of algorithms: constructive methods, algorithms based on local search and algorithms based on populations. Constructive methods generate a solution by the addition of components. Algorithms based on local search keep one reference solution at each iteration of the algorithm and find a better solution in the neighborhood of the reference one. Finally, population-based algorithms keep a set of solutions at each iteration of the algorithm and generate a new set of solutions considering the most relevant properties of the population. The most common techniques used in this third group form Evolutionary Computation (EC). Particularly, Evolutionary Algorithms (EAs) are a subset of EC in which the algorithms are inspired on Darwin's theory of biological evolution to generate new populations. The most common techniques to generate new solutions are selection, recombination and mutation. In particular, the Estimation of Distribution Algorithms (EDAs) [5] are a really promising type of algorithms.

EDAs are a class of EAs introduced by Mühlenbein and Paaβ [6]. The main characteristic of EDAs in comparison to the rest of EAs is that at each iteration EDAs learn a probability distribution from the selected solutions to "represent their main features" and to sample new solutions which are used to generate a new population. Recently, EDAs have been successfully used to solve permutation-based COPs by the use of probabilistic models specially designed for codifying probability distributions over permutation spaces [7]–[9]. Particularly, EDAs have been one of the main topics

recently studied in our research group *Intelligent Systems Group (ISG)* [10]–[12], among others. Unfortunately, the lack of theoretical analysis about these algorithms makes us ask ourselves why these algorithms have achieved excellent results on the studied problems. Moreover, due to the difficulty of the permutation spaces, there are no extensively developed studies which reinforce and support the use of these algorithms. With these two ideas in mind, the first phase of this thesis focused on the theoretical study of EDAs for permutation-based COPs.

Our starting point of the thesis has been the study of EDA which learns a Mallows model at each iteration of the algorithm. Due to the wide range of possible scenarios and behaviors that an EDA can manage by its definition, our study has assumed to study an EDA with infinite population to remove the randomness of the algorithm and to analyze it as a dynamical system. From this assumption, we have calculated the convergence behavior of our dynamical system. This analysis has been partially submitted [13], [14]. From these results, our next step would be studying the runtime analysis of the proposed EDA and generalizing the obtained results for several EDAs based on distance-based exponential models.

However, in the previously mentioned analysis, the obtained results have been very unexpected. For the simplest fitness functions, the algorithm cannot ensure to converge to the optimal solution. The convergence behavior of the algorithm is very dependent on the initial probability distribution used to run the algorithm and even if the search space is "small", the range of possible scenarios of the algorithm is still wide. Due to the outcome of the study, we have left this line of research and started a new one.

In the second phase of this thesis, we focus on the relations and matchings among COPs, COP instances and algorithms. The ideal goal is to find a "mathematical tool" such that given a COP instance the tool (oracle) returns us the best metaheuristic algorithm to solve that particular case. Obviously, the problem of finding these relations between instances and algorithms is as difficult as solving any COP instance with an algorithm in an efficient way. Nevertheless, in this field, researchers are paying attention to two main ideas: generating a taxonomy for COPs and for algorithms. We are more interested in the first classification.

The main problem of a taxonomization of COPs is that each problem has its own definition: that is to say, although two problems need the same number of elements to define one instance, the meaning of the elements is not the same. To face the obstacle, researchers have generated frameworks in which different COPs defined in the same search space can be represented, studied and compared, without considering the exact meaning of the elements.

The most popular technique to generate frameworks is the transformation of the fitness functions of several COPs. The most knowns are elementary landscape analysis, the Fourier transform over the permutation space and the Walsh transform over the space of binary sequences [15]–[19]. In this work, our motivation is the last mentioned transform.

The Walsh transform is a complete set of orthogonal functions introduced by Walsh in [20] and it has been used in the field of EC to solve many binary-based real-world COPs. Several examples of the use of the Walsh transform are to study the linkage discovery problem [21], [22], to create accurate surrogate models for black-box optimization [23]–[25] and to recover polynomials with significant small degree monomials [26], [27]. In all the mentioned examples, the authors design algorithms and methodologies to build or generate approximations of the fitness functions based on the Walsh transform and to extract the main features of the studied problem. However, we believe that the Walsh transform is able to give us more detailed information about COPs.

Once the background of the thesis has been presented, in the following sections the main objectives of this thesis, the proposed methodology to achieve them and the relevance of the research are described.

II. Objectives

In the first phase of this thesis, as mentioned in the introduction, the objective was to present a mathematical model to study EDAs based on distance-based exponential models for permutation-based COPs. In the previous studies developed in our research group, EDAs that have been studied learn Mallows models and Generalized Mallows models. The main reasons to choose the Mallows model were the following two: it is considered as the analogous distribution of the Gaussian distribution over permutation spaces, and its definition is dependent on a distance function defined over the permutation space which can generate many different scenarios, properties and results.

- **Objective I: Analysis of convergence behavior of EDAs based on Mallows models.** The mathematical tool proposed to achieve this objective was to use dynamical systems in order to remove errors produced by the randomness of the algorithm and to be able to recognize if the algorithm will converge to the optimal solution or which the attractive convergence points of the algorithm are. This objective has been partially submitted [13], [14].

In the second phase of this thesis, the main objective is to present and analyze the Walsh decomposition and to introduce a framework for binary-based COPs. From the presented framework, the idea is to compare and associate several binary-based COPs and to study the range of possible instances for each problem and their difficulty for the presented algorithms in the literature. The second main objective can be summarize in two points:

- **Objective II: Introduction of the Walsh decomposition and analysis and classification of binary-based COPs.** The goal is to identify the main features of each COP by the Walsh coefficients and compare them without considering their particular definition of the fitness function and representation of the solutions. The definition of the Walsh decomposition also shows us which instances can be generated from each problem. This objective has been partially submitted [28].

- **Objective III: Classification and study of the properties of the instances of several binary-based COPs by the Walsh decomposition.** The idea of this partial objective is to extract the main features to classify problem instances. The classification will help us to compute the boundary which differentiates between the polynomially solvable instances and the rest and to compute the number of local optimal solutions for each instance. From these results, an analysis of the matching of COP instances and algorithms is expected to be done.

III. Methodology

In this section, the proposed methodology to achieve the objectives is presented. For each objective, the followed tasks and steps are presented.

- **Objective I: Analysis of convergence behavior of EDAs based on Mallows models.**

I.1 Implementation of an EDA based on Mallows model and Generalized Mallows model and experimentation on the resolution of different problems.
I.2 Identification of a hierarchy of the studied scenarios, according to their complexity.
I.3 Obtaining convergence results of the algorithm by means of dynamical systems.
I.4 Writing of a paper with the obtained results and submission to a scientific journal.
I.5 Diffusion of the results in an international congress.

- **Objective II: Introduction of the Walsh decomposition and analysis and classification of binary-based COPs.**

II.1 Study of the properties and applications of the Walsh decomposition in the literature for COPs.
II.2 Calculation of the Walsh coefficients of several binary-based COPs .
II.3 Identification of a hierarchy of the studied problems according to their complexity and the relation among them.
II.4 Writing of a paper with the obtained results and submission to a scientific journal.
II.5 Diffusion of the results in an international congress.

- **Objective III: Classification and study of the complexity and properties of the instances of several binary-based COPs by the Walsh decomposition.**

III.1 Study of the definition, properties and relevance of the Walsh coefficients for binary-based COPs.
III.2 Classification of the instances of the studied binary-based COPs according to their difficulty.
III.3 Extraction of the features for polynomially solvable instances by the use of Walsh decomposition.
III.4 Writing of a paper with the obtained results and submission to a scientific journal.
III.5 Diffusion of the results in an international congress.

IV. Relevance

The field of optimization has always been very relevant in science. Any new improvement of a previous result is very well accepted by the community. Notwithstanding the great results that many metaheuristic algorithms have obtained in real-world problems, a better understanding of the problems and the algorithms helps to exploit their characteristics and to obtain a better performance overall. This improvement can be reflected as an increase of the profits or as a reduction of the total cost, for example.

At the beginning of this thesis, the chosen EDA to study in this thesis had several motivations which made a really interesting subject to study. First, the lack of theoretical studies over the permutation space. Secondly, the analogy of the Mallows distribution with the Gaussian distribution. Thirdly, the application of the existing mathematical modeling and the creation of new ones. Finally, the "use of the randomness to achieve deterministic results". All these motivations combined made this study very challenging and intriguing, and any obtained results from the study would show us when the studied algorithm is appropriate to solve a particular problem and its particularities. Even if the obtained results were not the expected ones at all, the methodology followed in the work [13] can be applied for any other EDA based on distance-based exponential models.

In the current objective of the thesis, it is logical that the main advantage of studying new frameworks to analyze and observe equivalences among all the possible fitness functions and COPs is to help researchers to generalize the particular results of each COP for the rest of problems without considering the specifications of each problem. Moreover, this analysis helps us to get closer to the goal of selecting the best algorithm to solve a particular COP instance. We believe that a deeper study in this field will directly propose novel ideas which differentiate from the classical scope. The analysis carried out by the Walsh decomposition will be as relevant as the ones performed by elementary landscape decomposition or the Fourier transform.

Acknowledgments

This research has been partially supported by Spanish Ministry of Science and Innovation through the projects PID2019-104966GB-I00/AEI/10.13039/501100011033, PID2019-104933GB-I00/AEI/10.13039/501100011033, PID2019-106453GA-I00/AEI/10.13039/501100011033 and BCAM Severo Ochoa accreditation SEV-2017-0718; and by the Basque Government through the program BERC 2018-2021 and the projects IT1244-19 and IT-1252-19; and by UPV/EHU through the project GIU20/054. Imanol holds a grant from the Department of Education of the Basque Government (PRE_2020_2_0166).

References

[1] C. R. Reeves, *Modern heuristic techniques for combinatorial problems. Advanced topics in computer science*. Mc Graw-Hill, 1995.
[2] F. Rothlauf, *Design of modern heuristics: principles and application*. Springer Science & Business Media, 2011.
[3] X.-S. Yang, *Engineering optimization: an introduction with metaheuristic applications*. John Wiley & Sons, 2010.

[4] M. R. Garey and D. S. Johnson, *Computers and Intractability; A Guide to the Theory of NP-Completeness*. New York, NY, USA: W. H. Freeman & Co., 1979.

[5] P. Larrañaga and J. A. Lozano, *Estimation of distribution algorithms: A new tool for evolutionary computation*. Springer Science & Business Media, 2002, vol. 2.

[6] H. Mühlenbein and G. Paaβ, "From recombination of genes to the estimation of distributions I. Binary parameters," in *International conference on parallel problem solving from nature*. Springer, 1996, pp. 178–187.

[7] B. Jarboui, M. Eddaly, and P. Siarry, "An estimation of distribution algorithm for minimizing the total flowtime in permutation flowshop scheduling problems," *Computers & Operations Research*, vol. 36, no. 9, pp. 2638–2646, 2009.

[8] S. Tsutsui, "Node histogram vs. edge histogram: A comparison of probabilistic model-building genetic algorithms in permutation domains," in *Evolutionary Computation, 2006. CEC 2006. IEEE Congress on*. IEEE, 2006, pp. 1939–1946.

[9] S. Tsutsui, M. Pelikan, and D. E. Goldberg, "Using edge histogram models to solve permutation problems with probabilistic model-building genetic algorithms," *IlliGAL Report*, vol. 2003022, 2003.

[10] J. Ceberio, E. Irurozki, A. Mendiburu, and J. A. Lozano, "A distance-based ranking model estimation of distribution algorithm for the flowshop scheduling problem," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 2, pp. 286–300, 2014.

[11] J. Ceberio, A. Mendiburu, and J. A. Lozano, "Introducing the Mallows Model on estimation of distribution algorithms," in *International Conference on Neural Information Processing*. Springer, 2011, pp. 461–470.

[12] E. Irurozki, "Sampling and learning distance-based probability models for permutation spaces," Ph.D. dissertation, University of the Basque Country, Donostia-San Sebastián, 2014.

[13] I. Unanue, M. Merino, and J. A. Lozano, "A mathematical analysis of edas with distance-based exponential models," in *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, 2019, pp. 429–430.

[14] ——, "A mathematical analysis of edas with distance-based exponential models," *Submitted to Memetic Computing*, 2021.

[15] F. Chicano, L. D. Whitley, and E. Alba, "A methodology to find the elementary landscape decomposition of combinatorial optimization problems," *Evolutionary Computation*, vol. 19, no. 4, pp. 597–637, 2011.

[16] J. Huang, C. Guestrin, and L. Guibas, "Fourier theoretic probabilistic inference over permutations." *Journal of Machine Learning Research*, vol. 10, no. 5, 2009.

[17] L. A. Christie, "The role of Walsh structure and ordinal linkage in the optimisation of pseudo-boolean functions under monotonicity invariance." Ph.D. dissertation, 2016.

[18] D. E. Goldberg, "Genetic algorithms and Walsh functions: Part I, a gentle introduction," *Complex systems*, vol. 3, no. 2, pp. 129–152, 1989.

[19] ——, "Genetic algorithms and Walsh functions: Part II, deception and its analysis," *Complex systems*, vol. 3, no. 2, pp. 153–171, 1989.

[20] J. L. Walsh, "A closed set of normal orthogonal functions," *American Journal of Mathematics*, vol. 45, no. 1, pp. 5–24, 1923.

[21] R. B. Heckendorn and A. H. Wright, "Efficient linkage discovery by limited probing," *Evolutionary computation*, vol. 12, no. 4, pp. 517–545, 2004.

[22] S.-S. Choi, K. Jung, and B.-R. Moon, "Lower and upper bounds for linkage discovery," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 2, pp. 201–216, 2009.

[23] S. Verel, B. Derbel, A. Liefooghe, H. Aguirre, and K. Tanaka, "A surrogate model based on Walsh decomposition for pseudo-boolean functions," in *International Conference on Parallel Problem Solving from Nature*. Springer, 2018, pp. 181–193.

[24] F. Leprêtre, S. Verel, C. Fonlupt, and V. Marion, "Walsh functions as surrogate model for pseudo-boolean optimization problems," in *Proceedings of the Genetic and Evolutionary Computation Conference*, 2019, pp. 303–311.

[25] D.-P. Yu and Y.-H. Kim, "On the effect of Walsh/Fourier transform in surrogate-assisted genetic algorithms," in *Proceedings of the 2020 Genetic and Evolutionary Computation Conference Companion*, 2020, pp. 235–236.

[26] O. Ocal, S. Kadhe, and K. Ramchandran, "Low-degree pseudo-boolean function recovery using codes," in *2019 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2019, pp. 1207–1211.

[27] E. Kushilevitz and Y. Mansour, "Learning decision trees using the Fourier spectrum," *SIAM Journal on Computing*, vol. 22, no. 6, pp. 1331–1348, 1993.

[28] I. Unanue, M. Merino, and J. A. Lozano, "A general framework based on walsh decomposition for combinatorial optimization problems," in *Proceedings of 2021 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, 2021.

Explainable condition monitoring from imprecise information.

Nahuel Costa
Computer Science Department
University of Oviedo
Gijón, Asturias, Spain
costanahuel@uniovi.es
Starting date: September 2020

Director: Luciano Sánchez
Computer Science Department
University of Oviedo
Gijón, Asturias, Spain
luciano@uniovi.es

***Abstract*—Condition monitoring refers to the process of controlling the information sources of a system in order to detect changes that may lead to failures or deterioration. This process takes place in a wide variety of fields, such as industry, healthcare or economics, and usually, regardless of the field of application, two main problems tend to arise. First, data are historical records that most of the time are subject to several problems that hinder its processing. Secondly, Machine Learning algorithms are used for this purpose but most of them act as black-box models. That is, based on some inputs, an output is predicted, leaving aside important questions such as why those decisions were made (and not others) or simply providing some insights into the nature of the data. For this reason, this thesis aims to contribute in this direction by applying Deep Learning models to monitoring data in order to provide an accurate and explainable diagnosis of the system to be monitored.**

***Index Terms*—Condition monitoring, data diagnostics, explainability, time series.**

I. Summary

There are many areas in which it is of interest to monitor certain data from which to extract knowledge for subsequent decision-making. This normally applied to industry, but nowadays the sources of information are very diverse, allowing this monitoring to be applied to problems such as detecting anomalies in the vital signs of a person in the ICU [1] or predicting the stock market [2]. Typically, we are dealing with data captured over time, which in the end can be regarded as time series. Precisely, when it comes to this type of data each case is unique and processing may differ considerably from one problem to another, which makes it hard to delve into time series research. This is compounded by the fact that shortcomings in data capture are quite common due to problems like sensor failure, susceptibility to noise, or simply lack of data.

Nevertheless, even if clean and ready-to-process data is available, Artificial Intelligence research has taken a path in recent years in which the main focus is established on neural networks applied mostly to Natural Language Processing (NLP) and Computer Vision, while the application of some of these powerful algorithms for time series is not yet fully exploited, where non-Deep Learning methods such as [3] still outperform sophisticated deep architectures.

Finally, despite being incredibly good at some tasks, most AI models behave as black boxes because they are based on feeding an input to an algorithm that outputs some number or class [4]. However, there are many fields where this is not enough and this is where the importance of explainability comes into play. It is crucial to know in some way how these models work internally in order to: 1) bring transparency to research and 2) provide certain interpretability of the models' decisions so that they can be easily used by people of any area outside AI.

A. Objectives

The main objective to be covered in the development of this thesis and its breakdown into sub-objectives is as follows:

- Design of explainable AI algorithms to provide tools for diagnosing inaccurate data.
  1) Contribute to fill gaps in time series processing.
  2) Use of Representation Learning mechanisms that provide information about the nature of the data, the relationship between different variables and their application to incomplete samples.
  3) Work on the interpretability and explainability of the latent variables learned by the models.
  4) Exploit generative models to simulate new data samples when there is not enough data.
  5) Development of learning mechanisms to model the learning of problems in which not all data is labeled.
  6) Provide open source code that allow the practical application of the above mechanisms.

II. Methodology and Work Plan

For better planning, the objectives have been structured into tasks to be performed during the thesis:

1) Literature review: study of the state of the art in explainable AI and imprecise data processing. Analysis of recent trends that can be applied to condition monitoring and diagnostic data problems as well as familiarization with the handling of low-quality data and identification of constraints.
2) Dig into generative models: generative models have shown remarkable results in a wide range of image

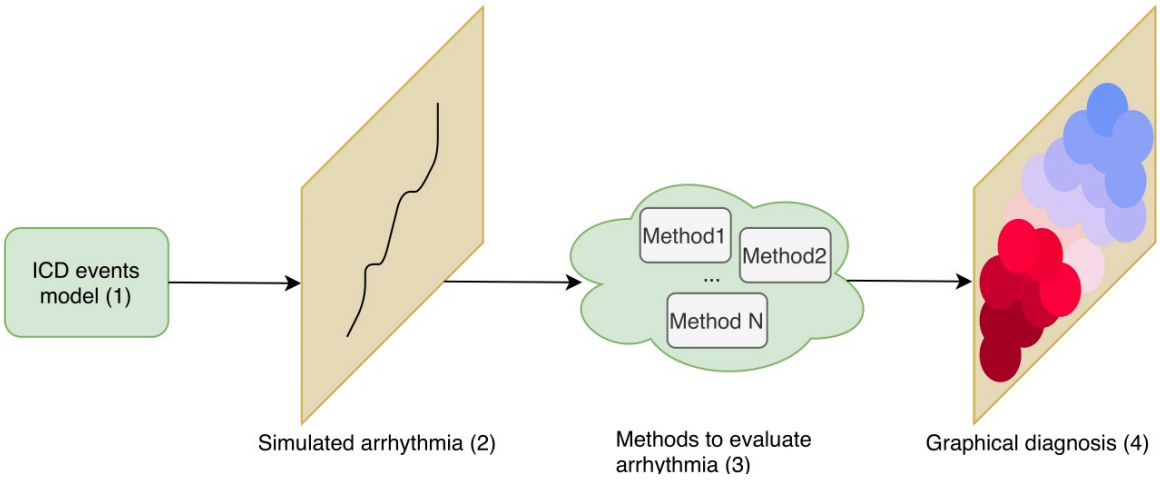


Fig. 1. Pipeline of the presented work in [1].

processing tasks. Mainly GAN and VAE networks are used in the literature to model complex data distributions from which to generate new data samples. This can be used for data augmentation in time series, especially in those cases where the number of samples is insufficient.

3) Latent variable exploration: Representation Learning mechanisms lie mostly on the internal representations learned by neural networks, known as latent spaces and are sources of information from which the relationship between input data can be explored.
4) Transfer learning applied to time series: time series, unlike other types of data such as words or images, represent completely different physical quantities (molecule concentration, heart monitoring, energy consumption, etc.). This makes it difficult to create pre-trained models that can serve as a basis for simpler models to transfer the obtained knowledge to fine-tune a particular problem. Therefore, the goal is to move in this direction in order to narrow this gap.
5) Development of lightweight models: computational resources, except in large companies, are generally limited, hence developing lightweight models that are easily deployable in devices with low computational capacities, such as mobile phones or raspberries, is highly valuable.
6) Search and share knowledge: translate the discoveries and contributions made into scientific communications as well as social posts and open code. Also, seek collaborations that enhance the research branches of each party.

III. Milestones achieved

As mentioned above, there are many fields susceptible to data monitoring. The following lines are intended to present a brief summary of the publications carried out to date, which deal with problems of very different natures but of great impact on society: the diagnosis of Atrial Fibrillation (AF) and the prediction of the Remaining Useful Life (RUL).

A. Atrial Fibrilation Diagnosis

The most common heart disease, Atrial Fibrillation (AF), is a degenerative condition in which the main symptoms, cardiac arrhythmias, progress from paroxysmal arrhythmia (spontaneous arrhythmia episodes) to permanent arrhythmia (uninterrupted episodes). This progression is a complex process that depends on several risk factors and an early diagnosis may condition the provision of optimal treatment. For those patients in latter stages of the disease, pacemakers are normally used to control the heart rate, along with continuous monitoring of the data collected by the device. This is where the main limitation arises: due to technical reasons the only information available are the dates and lengths of the recorded episodes. This is a drawback because patients with multiple episodes will probably be in the latest stages of AF, but the interest lies in patients with a short history (initial episodes) so that the worsening rhythm of AF can be anticipated. This goal is very challenging because it is difficult to find a model that fits such a small amount of data. Besides, the fact that the data is non-stationary makes the problem even more complicated, and this is exactly what we wanted to predict on the basis of a short sample, the transition from paroxysmal to permanent arrhythmias.

In [8] we provide a graphical analysis of the progression of AF using recurrent neural networks. Figure 1 summarizes the workflow followed:

- A generative model(1) is used to simulate real clinical data(2).
- The generated data is used for training different methods(3) to evaluate intracardiac records. Among these methods, further research is done to obtain a time series classifier based on adversarial training.
- A self-explanatory graphic map(4) is obtained when the proposed methods are fed with data from real patients with AF.

Getting medical data is a hard task due to privacy concerns,

that is why we developed a dynamical Markov model to simulate the behavior of real cardiac arrhythmias, and thus generate a dataset that could reflect the different stages of AF. Even though this is a crucial component, the main novelties are the use of discriminator networks of GANs (Generative Adversarial Networks) [5] and the visual projection of the activations of the neurons of the models used.

To begin with, GANs consist of 2 networks: a generative net and a discriminative net. The generator net produces new data instances from noise, while the discriminator receives real data along with data from the generator and decides whether the generator's data belongs to the same distribution as the real data. From this verdict, the parameters of both networks are adjusted to improve in the next iteration until the generator is able to produce realistic data.

GANs are a hot research topic and are widely used in image processing for several tasks [6] but seldom are applied to time series, not to mention intracardiac data. However, if a GAN is trained with arrhythmias of different classes, a discriminator capable of detecting arrhythmias of that type is obtained and that is the key idea behind this work. Using a customized recurrent version of a GAN (to deal with temporary data) we train multiple models with different types of arrhythmia. This way we obtained a set of discriminators specialized in recognizing very specific arrhythmia types, a property that is exploited to build an ensemble of discriminators that, given a patient's data, predict the class of arrhythmias that best reflect the patient's condition.

We found that the ensemble is highly responsive when the arrhythmia is paroxysmal and this is the desired result because these are the cases with clinical interest. Misclassifications arise from patients in permanent arrhythmia, however, the evolution of the patient is evident in those cases while the concern lies in anticipating that state so that specialists can act prematurely to avoid possible future complications. We train the models with simulated data obtained from the Markov model and tested them with labeled data from pacemakers and defibrillator systems provided by the HUCA (Hospital Central de Asturias).

Finally, to provide a graphical tool to analyze the predicted results we project a visualization of the activation of the last layer neurons as can be seen in Figure 2. The horizontal axis is labeled $beta$ which can be understood as the expected number of days between two AF episodes. The vertical axis is labeled α and measures the speed of the progression. The lower the value of α the quickest the progression to permanent AF. Red areas are the highest activations, and blue areas the lowest. In the case of this patient the red region is centered in α= 0.9994 and $beta$= 180. This means that the patient most likely started to suffer AF episodes every 6 months (180 days), although the evolution of the arrhythmia is moderate (high α value) and no risk is expected in the near future.

The results achieved in this work meet objective 4) and task 2) as two generative models were used: a Markov model to simulate the behavior of Atrial Fibrillation in order to build a consistent dataset and a GAN. In the latter, although the GAN networks were not used for generating data, they indeed can effectively generate time series data, nevertheless, the discriminator was used instead to build an ensemble classifier.

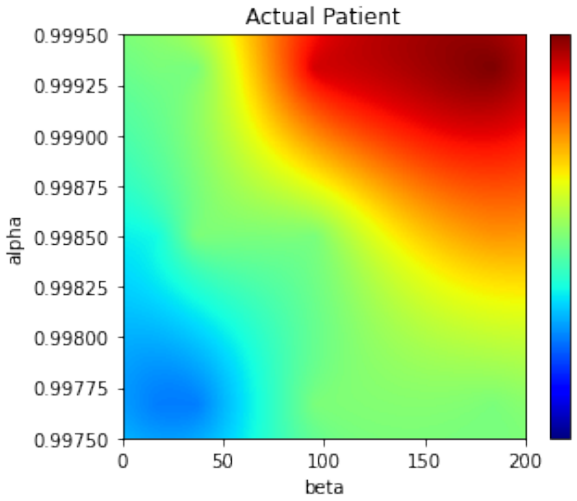


Fig. 2. GAN ensemble projection of an actual patient.

B. *Semi-supervised recurrent Variational Autoencoder*

Autoencoders are a family of neural networks which follow an encoder-decoder scheme and have the ability to learn in an unsupervised way a simplified representation of the data, the so-called latent space. Variations of the original model have been developed in order to enhance classification and clustering tasks until the emergence of Variational autoencoders (VAEs) [7], whose main purpose is the generation of new data.

In a VAE the encoder learns to compress the data into the latent space from which the decoder can reconstruct it to its original dimension as conventional autoencoders do. What makes VAE different is that they are rooted in Bayesian inference, which lets the latent space be described by the parameters learned by the encoder that initialize the probability distribution to which the data belongs, so that the decoder can not only reconstruct the input data, but can also generate new samples from that distribution. This causes the inputs not to be encoded in a set of fixed vectors, but to a continuous latent space instead, as the compression depends on a probability distribution. This means that the data is organized according to its similarities and thus, if two close points in the latent space were decoded, they should give similar contents. Precisely, other unsupervised techniques such as clustering algorithms lack this property. Although they do prioritize grouping data of a similar nature, the visual disposition of the clusters can often be arbitrary. On the other hand, neither can the VAE latent space be used for clustering since the encoded data tend to be overlapped to prioritize the generative process.

In [9] we propose a methodology for time series data capable of combining the above properties, that is, depicting the input data into clusters, while preserving a continuous representation according to its underlying complexion. Figure

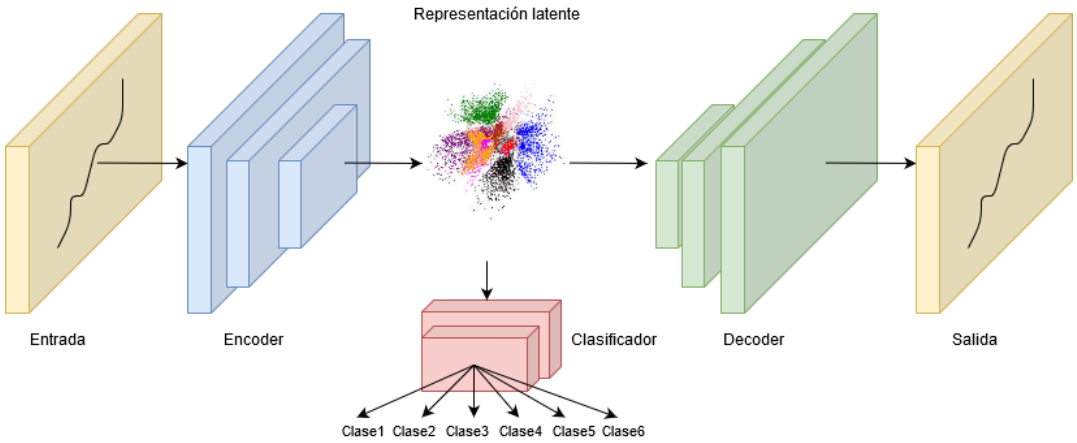


Fig. 3. Pipeline of the presented work in [2].

3 illustrates this idea: A VAE is trained with time series data to learn in the latent space a representation where similar data is located in similar areas while different data is located apart. A classifier is introduced in the training process to control the regulation of the latent space and thus preventing the mentioned overlapping. A latent space of these characteristics serves as a basis for encoding new data into the latent space for establishing a diagnosis based on their proximity to nearby points. It is noteworthy to mention that, although VAEs are widely used in image processing, there are hardly any works adapted to time series [10] [11], that is why the model we present is a novel implementation based on recurrent networks, namely bidirectional LSTMs to capture the temporal dependencies of the data.

To test the effectiveness of the proposed framework we used it to address the diagnosis of Atrial Fibrillation (AF). We demonstrate that our framework outperforms state-of-the-art classifiers for time series for the case at hand. Besides, it should be noted that the baseline methods we present do not include any representation mechanisms, but simply predict the class to which each sample belongs which makes us appreciate the importance of a visual interpretation of the data.

Figure 4 presents the learned latent space for the training data along with the compressed representation of an actual patient (red dot). Remember, α measures the speed of progression of arrhythmias and values close to 0.999 indicate a slow progression of AF. β indicates the average time between arrhythmias. This case is located in the cluster with parameters α = 0.998 and β = 30. Firstly, the parameter β is closer to 30, but due to its proximity to the lower-left group (β = 180), it can be understood that its evolution is on the way to reach 30, possibly a value between 180 and 30. Secondly, the most critical parameter, α, corresponds to a value of 0.998, which means that the evolution is closer to a permanent arrhythmia. This is not the most critical case, but it may need medical intervention in order to prevent potential complications.

It should be emphasized that both in [8] and [9] we offer different solutions to the same problem applying generative networks, not to generate data, but to diagnose it. The map in Figure 4 is build with the latent space offered by the VAE rather than with the activation of internal neurons. It is preferred because it enhances the fact that similar data is clustered in similar areas therefore even if the prediction is not exactly correct it offers a perception of what it might be based on its proximity to other points whose diagnosis is known.

The contributions made here completely meet the objective 2) and task 3). The presented map is entirely based on the VAE´s learned latent space. We demonstrated that it can provide information about the nature of the data and the relationship between different variables.

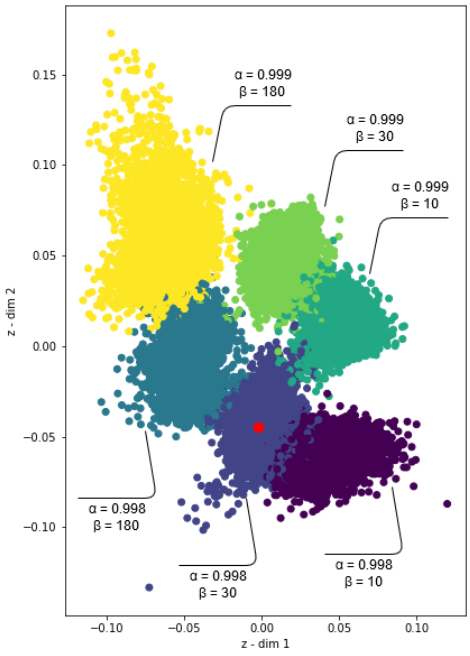


Fig. 4. Projection of arrythmias of an actual patient.

C. RUL estimation

Engineering maintenance and prognostics are a must in modern aircraft engines. Data is routinely collected from the engine to monitor and prevent it from operating in undesirable conditions. The knowledge of the system built into the engine and aircraft is configured to trigger alerts that highlight the need for pilot action, maintenance action or directly shut the engine down if a significant condition is encountered. Over the years, the number of variables and data collected has increased substantially which is positive in terms of making more accurate diagnostics, but at the same time increases the difficulty to reach them since traditional strategies such as corrective maintenance of failures and scheduled preventive maintenance are becoming less capable of meeting the growing industrial demand for efficiency and reliability. On the other hand, smart Prognostics and Health Management (PHM) technologies are showing promising capabilities for application in industries. Remaining useful life (RUL) is a key metric in this regard and can be estimated from the historical data that sensors record on each trip, which is very important to improve maintenance schedules and avoid engineering, safety and reliability failures and, as a consequence, determine engine deterioration, increase engine flight time and reduce maintenance costs.

The main limitation of most approaches for RUL estimation is that they require extensive prior knowledge about the physical systems that is usually not available in practice. This is precisely why it is so important to develop data-driven approaches capable of modeling degradation characteristics based solely on historical sensor data. In this way, knowledge can be generated from the collected data with little prior prognostic experience to infer valuable system information, such as RUL.

Remember that our goal is to extend the range of application of our findings to different fields. Diagnosis of AF and modeling of aircraft engine deterioration, although seemingly unrelated problems, are actually quite similar (engine breakpoint $\approx$ transition from paroxysmal to permanent arrhythmia). Therefore, in this sense we continue with the explainability of latent spaces for time series in "Remaining Useful Life Estimation Using a Recurrent Variational Autoencoder" [paper under revision]. We present a modification of our recurrent VAE [9] applied to EHM data from the popular NASA´s C-MAPSS datasets. In addition, we validate the algorithm with real Turbofan engines from Rolls-Royce, as RUL estimation is an actual engineering problem posed by this company and generally a trending topic in the field [12]. We discovered that discarding the decoder, i.e., having an encoder-based architecture that learns to compress data to a continuous latent space driven by the variational inference, enhances the learned representations. Thus, the training is divided into two phases: first, a pre-training phase where the learned high-dimensional features are projected into the latent space is done. Consequently, that knowledge can be used to fine-tune a simple linear model for a particular problem, either classification or regression. In this case, regression to estimate the RUL of aircraft engines. We demonstrate that this method outplays state-of-the-art models for RUL prediction, furthermore providing an explainable diagnosis of the data, since, as we did in [9], new data is projected into the latent space to give insight into the actual deterioration of the system. This information can be used by the mechanics to make a decision regarding the aircraft monitoring, either to make it more exhaustive or to take the vehicle to the workshop for a more complete check-up, thus anticipating the breakpoint before failure and expanding the lifetime of the engines.

This work is again aligned with task 3) but also to some extent with task 5). The lightness of the model and its recurrent nature would allow incorporating the model as a diagnostic system on any hardware with limited computational capabilities. What is more, it also allows updating the learned patterns each time a new data sample becomes available, thus coping with the non-stationarity of the data distribution.

IV. Future Work

In NLP data are words stored sequentially, something similar to time series, hence the usual way of dealing with this type of data was also with recurrent networks. However, in 2017 a new concept emerges: Transformers. A new architecture based on attention is proposed in [13]. Attention is a mechanism first presented in [14] to represent the most relevant information in a vector . It is quite simple to understand if we think about the way we translate into another language: we do not look at the whole sentence for each word being translated, but focus on specific words in the original sentence to translate the current word, i.e., we fix our attention on specific parts of the sentence.

The advantages that transformers have over RNNs are the following: 1. They have more memory, therefore they do not suffer from short-term memory. 2. They can be parallelized because data is not processed sequentially.

Since its publication, advances in NLP have been and still are oriented almost exclusively to Transformers and they are also beginning to stand out in Computer Vision. However, although it is also ideal for time series, research is limited to a few recent papers [15] [16] tailored to multi-step forecasting. This is where BERT arises: BERT is an NLP model presented by Google in [17] based on Transformers. Transformers have an encoder-decoder architecture, but it is the encoder that learns the information representations from which other tasks can be done. Under this premise what they do is to pre-train the Transformer encoder with data from many different sources and offer the model so that the learned knowledge can be leveraged to fine-tune for other problems.

In NLP it is feasible to offer a generic model from which to apply transfer learning to a particular problem because text, after all, has the same nature regardless of its origin but in time series the same does not happen. The physical composition of each problem can vary completely and that makes it difficult to pre-train a model with data from several different sources.

However, this assumption is not absolute and some works have applied transfer learning for time series [18] and this

is the line of research on which we are currently focusing. Being able to develop a framework similar to BERT, from which the knowledge generated could be leveraged for almost any problem, would greatly facilitate entry into time series research.

On the other hand, after hundreds of publications claiming that "attention is all we need", new ideas are emerging that break this assertion. The attention layer, although better than RNNs is still a bottleneck for network efficiency. That is why Google researchers replaced BERT's attention layers with Fourier transforms and report reaching 92% accuracy of BERT but 7 times faster [19]. Also, in [20] the authors propose an MLP-based architecture that improves Transformers in some NLP and vision tasks.

It is clear that research in Transformers is following an exponential pace and these recent advances can mean significant breakthroughs for time series processing as well.

Finally, there is another trend that is achieving significant results in Computer Vision known as Contrastive Learning. The main idea of Contrastive Learning is to learn, in an unsupervised way, representations so that in the latent space similar samples stay close to each other, while different samples are distant, something quite similar to what we achieved with VAEs. The main motivation for this mode of learning comes from the learning patterns we humans have. We recognize objects without remembering all the small details, for instance, we find it easy to look at a picture and find a chair in it based on color, shape and some other features. Roughly speaking, we create some kind of representations in our mind and then use them to recognize new objects. It would be of interest to be able to apply this learning pattern to our domain to provide a better understanding of the data or to use it in problems where labels are missing, just to name a few possible applications.

These two streams would fit with tasks 3, 4 and 5. Contrastive learning is pretty much along the lines of what has been achieved with VAE, but we strongly believe that its results are worth investigating for non-image domains. Transformers are the ideal component for developing lightweight models that deal with time-dependent data because they can be parallelized and even with improvements such as [19] together with transfer learning they could even be trained on CPU, which is unheard of for large Deep Learning models.

V. Relevance

The contributions of this research are extensive. First, we want to contribute to making the development and application of Deep Learning models for time series as simple and intuitive as possible, thus attracting researchers from other fields and reducing the gap with respect to NLP and Computer Vision. For this, it is key to develop explainable tools that allow to understand what the algorithms do internally and also to provide an interpretability of what models learn.

As a consequence of the development of the field, the advances developed will eventually be transferable to society. A glimpse of this objective can be seen in the publications mentioned above. Despite being research works, their purpose is to offer diagnostic tools for a wide range of problems. Especially to the diagnosis of AF and aircraft engines, allowing in the first case to improve the quality of life of patients and in the second to enhance the savings of companies by extending the useful life of aircraft.

References

[1] Saeed, M., Lieu, C., Raber, G., & Mark, R. G. (2002, September). MIMIC II: a massive temporal ICU patient database to support research in intelligent patient monitoring. In Computers in cardiology (pp. 641-644). IEEE.

[2] Chen, K., Zhou, Y., & Dai, F. (2015, October). A LSTM-based method for stock returns prediction: A case study of China stock market. In 2015 IEEE international conference on big data (big data) (pp. 2823-2824). IEEE.

[3] Lines, J., Taylor, S., & Bagnall, A. (2018). Time series classification with HIVE-COTE: The hierarchical vote collective of transformation-based ensembles. ACM Transactions on Knowledge Discovery from Data, 12(5).

[4] Castelvecchi, D. (2016). Can we open the black box of AI?. Nature News, 538(7623), 20.

[5] Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2014). Generative adversarial networks. arXiv preprint arXiv:1406.2661.

[6] Gui, J., Sun, Z., Wen, Y., Tao, D., & Ye, J. (2020). A review on generative adversarial networks: Algorithms, theory, and applications. arXiv preprint arXiv:2001.06937.

[7] Kingma, D. P., & Welling, M. (2013). Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114.

[8] Costa, N., Fernández, J., Couso, I., & Sánchez, L. (2020). Graphical Analysis of the Progression of Atrial Arrhythmia Using Recurrent Neural Networks. International Journal of Computational Intelligence Systems, 13(1), 1567-1577.

[9] Costa, N., Sanchez, L., & Couso, I. (2021). Semi-Supervised Recurrent Variational Autoencoder Approach for Visual Diagnosis of Atrial Fibrillation. IEEE Access, 9, 40227-40239.

[10] Park, D., Hoshi, Y., & Kemp, C. C. (2018). A multimodal anomaly detector for robot-assisted feeding using an lstm-based variational autoencoder. IEEE Robotics and Automation Letters, 3(3), 1544-1551.

[11] Suh, S., Chae, D. H., Kang, H. G., & Choi, S. (2016, July). Echo-state conditional variational autoencoder for anomaly detection. In 2016 International Joint Conference on Neural Networks (IJCNN) (pp. 1015-1022). IEEE.

[12] Li, X., Ding, Q., & Sun, J. Q. (2018). Remaining useful life estimation in prognostics using deep convolution neural networks. Reliability Engineering & System Safety, 172, 1-11.

[13] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. arXiv preprint arXiv:1706.03762.

[14] Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473.

[15] Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., & Zhang, W. (2020). Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting. arXiv preprint arXiv:2012.07436.

[16] Lim, B., Arik, S. O., Loeff, N., & Pfister, T. (2019). Temporal fusion transformers for interpretable multi-horizon time series forecasting. arXiv preprint arXiv:1912.09363.

[17] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

[18] Fawaz, H. I., Forestier, G., Weber, J., Idoumghar, L., & Muller, P. A. (2018, December). Transfer learning for time series classification. In 2018 IEEE international conference on big data (Big Data) (pp. 1367-1376). IEEE.

[19] Lee-Thorp, J., Ainslie, J., Eckstein, I., & Ontanon, S. (2021). FNet: Mixing Tokens with Fourier Transforms. arXiv preprint arXiv:2105.03824.

[20] Liu, H., Dai, Z., So, D. R., & Le, Q. V. (2021). Pay Attention to MLPs. arXiv preprint arXiv:2105.08050.

MORK: Metaheuristic Optimization framewoRK

Raúl Martín Santamaría
Universidad Rey Juan Carlos
Móstoles, Madrid, Spain
raul.martin@urjc.es

Datos de la Tesis Doctoral

- Tutores: José Manuel Colmenar Verdugo (josemanuel.colmenar@urjc.es) y Abraham Duarte (abraham.duarte@urjc.es).
- Departamento: Ciencias de la Computación, Arquitectura de Computadores, Lenguajes y Sistemas Informáticos y Estadística e Investigación Operativa.
- Afiliación: Universidad Rey Juan Carlos.
- Fecha de inicio: 2019.

***Resumen* — Los algoritmos heurísticos y metaheurísticos son una herramienta fundamental para la creación de aproximaciones que generen buenas soluciones en tiempos contenidos. Aunque a diferencia de las estrategias exactas no pueden demostrar que una solución sea óptima, su principal ventaja radica en poder resolver instancias que serían imposibles para algoritmos tradicionales. En este trabajo, se presenta el *framework* de desarrollo *MORK*, con el objetivo de proporcionar un marco común para el desarrollo de aproximaciones heurísticas, de forma fácil, cómoda y ayudando a evitar los errores más comunes. El *framework* se ha aplicado con éxito a diferentes problemas de investigación, no solo del autor sino por terceras personas ajenas al proyecto.**

***Palabras clave* —** heurísticas, optimización, combinatoria, marco de desarrollo, metaheurísticas.

I. Introducción

En esta sección se introducen, primero, los conceptos clave relevantes del proyecto en curso, para luego explicar brevemente la propuesta.

A. Conceptos clave

Para poder garantizar la optimalidad de una solución en el contexto de un problema de optimización, se suele recurrir a algoritmos exactos, dentro de los cuales se encuentran implementaciones comerciales, como CPLEX [2] o Gurobi [6], o adaptaciones de algoritmos tradicionales como programación dinámica (*dynamic programming*) o ramificación y poda (*branch and bound, B&B*).

La principal limitación de este tipo de aproximaciones es la escalabilidad: en muchos problemas, al aumentar los tamaños de las instancias a resolver de forma lineal, el esfuerzo computacional necesario para calcular la solución óptima aumenta de forma exponencial. A esta clase de problemas, para los que no existen algoritmos exactos con complejidad polinómica, se les denomina problemas NP-Duros.

Una alternativa a los algoritmos exactos consiste en el uso de estrategias heurísticas o metaheurísticas, que aunque no pueden garantizan la optimalidad de la solución, consiguen muy buenos resultados en tiempos de cómputo muy acotados. Por ello, la aplicación de algoritmos heurísticos y metaheurísticos es clave en el contexto de los problemas NP-Duros, siendo a menudo la única forma de resolver instancias de tamaño razonable en tiempos de cómputo asequibles para las máquinas de hoy en día.

Encontrar estrategias que generen buenas soluciones en tiempos reducidos es uno de los grandes focos de la investigación actual [5] [4].

B. Motivación y estado del arte

Como se adelantaba en la sección anterior, una de las estrategias más recurridas para resolver problemas NP-Duros consiste en el uso de estrategias heurísticas y metaheurísticas. Durante el desarrollo de estrategias para diferentes problemas, suele surgir una dicotomía: o bien utilizar un resolutor de modelos exactos, comúnmente denominado *solver*, o bien implementar parcial o completamente toda la funcionalidad algorítmica necesaria.

Cada aproximación tiene sus ventajas y sus inconvenientes, dependiendo del caso de uso. Por una parte, una de las principales ventajas de la primera aproximación puede ser la reducción de la carga de trabajo, es decir, reutilizar componentes reduce el tiempo de desarrollo y por lo tanto el tiempo hasta obtener retroalimentación sobre la aproximación utilizada. Por otra parte, una de las mayores desventajas de esta aproximación es que el proyecto resultante puede ser muy rígido, es decir, dificulta realizar adaptaciones futuras. La ventaja de que un *solver* genérico pueda aplicarse a cualquier problema supone a su vez un inconveniente: realizar adaptaciones u optimizaciones específicas para un problema concreto puede ser una tarea extremadamente compleja, comparada con la segunda aproximación. Además, la curva de aprendizaje de algunas librerías puede dificultar su integración.

Entre las librerías más importantes relacionadas con estrategias heurísticas y metaheurísticas, cabe destacar:

- jMetal [11]: Se trata de una librería de algoritmos poblacionales de código abierto para problemas, fundamentalmente, multi-objetivo. El sistema de optimización utiliza listas de variables, restricciones y objetivos para generar buenas soluciones para un problema dado.
- Optaplanner [3]: se trata de un motor de resolución de código abierto basado en restricciones (o en inglés, *AI constraint solver*). Implementa tanto algoritmos exactos como heurísticos. El usuario modela la solución y las restricciones y detalla como se realiza el cálculo de la

función objetivo, y Optaplanner genera soluciones para el problema utilizando diferentes aproximaciones.

- Global Optimization Toolbox [10]: Software de pago por *Mathworks*. De forma similar a Optaplanner, permite definir una función objetivo junto a una serie de restricciones. El *software* utiliza múltiples aproximaciones metaheurísticas para generar soluciones factibles.

Sin embargo, como se ha mencionado anteriormente, el uso de estas librerías supone varios inconvenientes. Tanto Optaplanner como jMetal ponen restricciones sobre el modelado de las soluciones. Implementar una función objetivo que pueda evaluarse de forma incremental es una tarea compleja y propensa a errores, por lo que afinar la aproximación para conseguir un buen rendimiento puede ser una tarea extremadamente compleja. jMetal proporciona algoritmos metaheurísticos, especialmente de tipo poblacional, pero no da un marco de desarrollo común para cualquier tipo de aproximación metaheurística.

Por otra parte, la segunda aproximación, en la que se desarrolla completamente el algoritmo, supone un gran esfuerzo para cada problema, pero tiene la ventaja de que al crear código específico para cada problema, hace que sea fácil realizar optimizaciones específicas para el mismo. Sin embargo, esta aproximación crea una gran cantidad de código redundante (lectura de instancias, exportación de resultados a diferentes formatos, algoritmos comunes, etc). Muchos investigadores que optan por esta estrategia acaban realizando sus propios repositorios de código común donde se implementa funcionalidad genérica para poder aplicarla más fácilmente otros problemas.

La propuesta presentada en este trabajo es un híbrido entre las dos aproximaciones presentadas anteriormente. Por un lado, implementa una gran parte de la funcionalidad que se considera común entre diferentes proyectos, y por otra, ofrece implementaciones de algunos de los algoritmos metaheurísticos más utilizados (GRASP [12], VNS [7], VND [8], etc). Su principal ventaja radica en su diseño modular y extensible, permitiendo a los usuarios elegir qué funcionalidad concreta quieren utilizar, extendiendo, personalizando o creando nuevas componentes según va siendo necesario, de forma fácil y sencilla.

Revisando la literatura no se ha encontrado ninguna librería de desarrollo integral que cubra desde el proceso de tratamiento y validación de instancias hasta la generación y análisis de resultados, facilitando el proceso de análisis de resultados al generar diversas métricas comunes de forma automática. Con este propósito, se presenta en este trabajo el *framework* de desarrollo metaheurístico *Mork*.

El objetivo de *Mork* es proporcionar un marco de desarrollo común que facilite en la medida de lo posible el desarrollo de estrategias heurísticas. No es un reemplazo para librerías existentes como jMetal, puesto que se pueden integrar fácilmente algoritmos externos (ver sección de trabajo futuro para más detalles). Gracias a su diseño modular, proporciona diversas características que pueden ser utilizadas independientemente del resto de funcionalidad. Cabe destacar que, a diferencia de Optaplanner o jMetal, no existe un modelo de solución concreto que sea necesario utilizar. Por el contrario, el usuario del *framework* tiene total libertad para definir las estructuras de datos de la solución y las operaciones que sobre ella se pueden aplicar.

II. Metodología y plan de trabajo

En esta sección se explica en detalle la propuesta, las características de la misma y su funcionamiento, y un resumen de implementación de un caso de uso por parte de un usuario investigador.

A. Arquitectura y requisitos

El diseño de la plataforma o *framework* de desarrollo es completamente modular, de forma que el uso de la mayor parte de las componentes sea opcional, ofreciendo una implementación básica por defecto que puede ser desactivada o sustituida por una específica por el investigador para cada problema concreto. El lenguaje de programación elegido para crear el *framework* es Java 11, debido a su equilibrio entre comodidad y velocidad, la compatibilidad con múltiples sistemas operativos y múltiples lenguajes, y la existencia de numerosas librerías programadas en Java que podrían integrarse fácilmente con la propuesta.

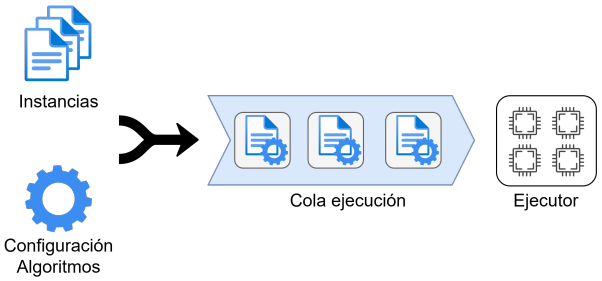


Fig. 1. Sistema de ejecución interno. Los ejecutores son una capa de abstracción sobre los recursos a usar, pudiendo ser estos de un solo hilo, múltiples CPUs en un mismo ordenador o un sistema distribuido.

Específicamente, el funcionamiento simplificado de la propuesta puede resumirse con la Figura 1. Primero, las instancias se cargan y opcionalmente validan. Para cada combinación formada por cada instancia, algoritmo e iteración, se crea una unidad de ejecución que es enviada a una cola, donde posteriormente será despachada a un ejecutor. Un ejecutor no es más que una abstracción para maximizar el uso de los recursos disponibles, minimizando el tiempo de ejecución, acorde a las restricciones del usuario. Ejemplos de implementaciones de un ejecutor pueden ser un hilo de CPU, varias CPUs en un mismo ordenador, o un clúster de cómputo distribuido. Para cada tupla (instancia, algoritmo, iteración), se lleva a cabo el proceso detallado en la Figura 2, el cual consiste en ejecutar el algoritmo y sus diferentes fases para generar un resultado dada una unidad de trabajo. Destacar que este diseño de unidades de trabajo hace que la ejecución sea fácilmente paralelizable. Los resultados se agregan de forma automática al finalizar.

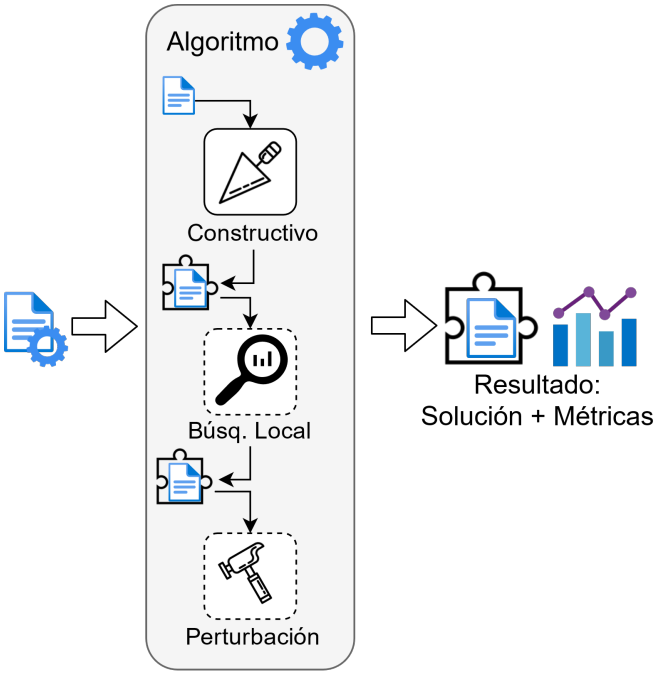


Fig. 2. Ejemplo de unidad de trabajo con un hipotético algoritmo con tres componentes: constructivo, búsqueda local y perturbación. En línea discontinua las componentes opcionales.

Entre las principales características que ofrece para los investigadores, cabe destacar: la reproducibilidad, la reusabilidad de los componentes y la velocidad de desarrollo.

1) Reproducibilidad de los experimentos: Utilizando, por una parte, contenedores (implementación actual basada en *Docker*) para facilitar la ejecución de experimentos en cualquier ordenador fácilmente. Por otra parte, utilizando un generador de números aleatorios personalizado de forma que se garantice la reproducibilidad de los experimentos bajo la misma semilla aleatoria en entornos paralelos.

2) Reusabilidad de los componentes: Al delegar las funcionalidades clave en diferentes componentes, conseguimos aislar el funcionamiento de las mismas, comunicándose entre ellas utilizando interfaces bien definidas, de forma que sustituir, extender y reusar componentes entre proyectos sea una tarea trivial.

3) Velocidad de desarrollo: Al proveer de una implementación básica por defecto para la mayoría de las componentes clave, pero extremadamente fácil de extender y modificar, aceleramos el desarrollo, siempre dando la opción al investigador de adaptar el *framework* a sus necesidades reales.

B. Caso de uso

En esta sección se resume, desde la perspectiva del usuario investigador, el procedimiento necesario para resolver cualquier problema de optimización mono-objetivo utilizando la propuesta presentada. La Figura 3 muestra este procedimiento, que se detalla a continuación.

Primero, es necesario modelar tanto la instancia como la solución. Para ello, es suficiente con extender los componentes `Instance` y `Solution` respectivamente. En dichas clases, el investigador implementará las estructuras de datos que considere apropiadas para modelar el problema, y los métodos necesarios para calcular el valor de la función objetivo de la misma.

Opcionalmente (marcado con línea discontinua), el siguiente paso sería implementar un *parser* de entrada, en caso de que el formato sea específico y no un estándar como JSON o CSV. Además, se pueden implementar validadores para verificar que todas las instancias están bien formadas y las soluciones son válidas en toda la ejecución de la fase de experimentos.

Tras ello, el siguiente paso sería implementar al menos un método constructivo, es decir, dada una instancia, generar una solución válida para el problema. Las soluciones no tienen por qué tener una calidad elevada, pero deben ser válidas dentro de las restricciones del problema. Opcionalmente, se pueden implementar vecindarios para sean utilizados, por ejemplo, en búsquedas locales, extender algoritmos existentes en la librería o implementar nuevos.

Una vez que tenemos las componentes de nuestra aplicación listas, pasamos a definir un experimento, que viene a ser uno o más algoritmos a ejecutar con diferentes configuraciones para un conjunto de instancias dado. Normalmente se usa un conjunto de instancias preliminar para acelerar la fase de experimentación preliminar y no obtener *overfitting* [9] en el ajuste de parámetros del algoritmo.

Finalizado el diseño del experimento, el último paso sería analizar los resultados. En caso de ser necesario, corregir los errores que el *framework* haya detectado durante la fase de experimentación, y proponer nuevas aproximaciones hasta tener buenos resultados.

C. Funcionalidad avanzada

El *framework* implementa la siguiente funcionalidad avanzada que puede ser utilizada por el usuario investigador tras activarla simplemente modificando el parámetros de configuración correspondiente.

1) Generación personalizada de números aleatorios: Para garantizar que los experimentos sean reproducibles, se provee de una clase especial generadora de números aleatorios. No es suficiente con que una clase dada funcione en entornos paralelos, sino que debe garantizar que los experimentos sean reproducibles independientemente del tipo de ejecutor empleado. Si no, cada ejecución podría generar resultados diferentes, debido, entre otros, a posibles variaciones de asignaciones de recursos a diferentes hilos de ejecución.

2) Paralelización automática: Aunque se pueden implementar opcionalmente algoritmos paralelos, *Mork* permite paralelizar a nivel de la tupla (instancia, algoritmo, iteración) de forma automática a petición del usuario, teniendo en cuenta el tiempo individual. En otras palabras, al utilizar el sistema de paralelización automática, cada ejecución individual dura lo mismo, pero el tiempo de experimentación total se reduce en un factor igual al número de núcleos utilizados.

3) Validación integrada de instancias y experimentos: El usuario define qué condiciones tiene que cumplir el modelo

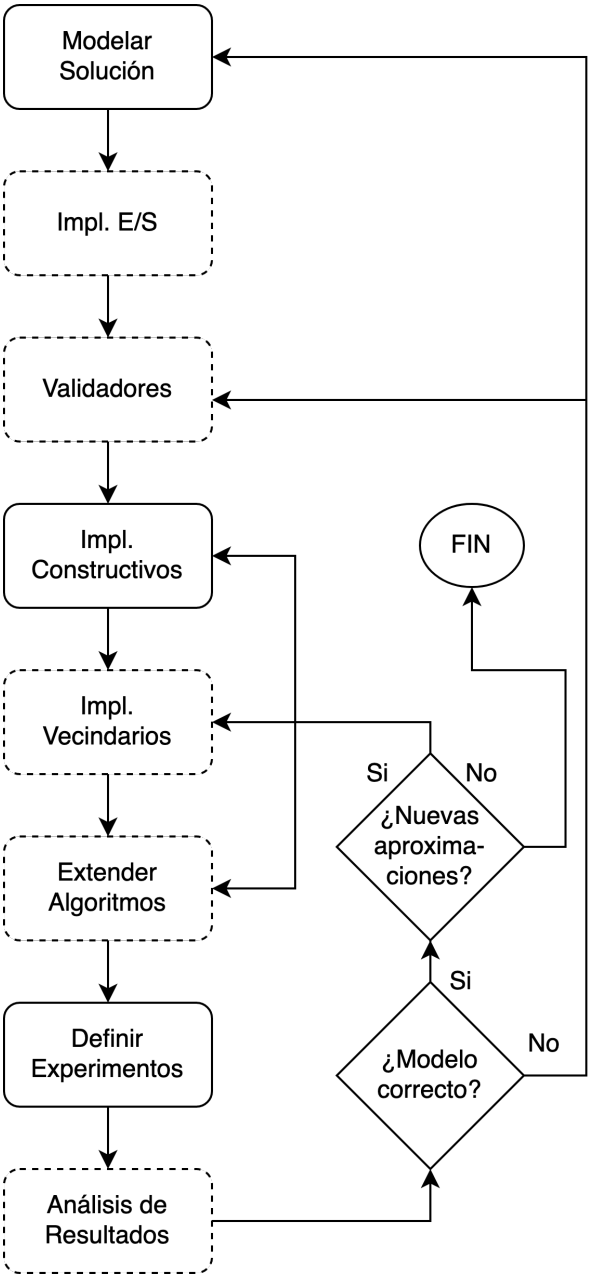


Fig. 3. Ejemplo de funcionamiento a nivel de usuario. Se representa mediante borde discontinuo aquellos elementos cuya implementación es opcional, y con línea continua las componentes o pasos que debe realizar el usuario.

del problema en diferentes puntos de la ejecución de los experimentos, y el *framework* se encarga de verificar su correcto funcionamiento, informando de inconsistencias en caso de producirse en cualquiera de los componentes. La detección temprana automática de posibles bugs minimiza el tiempo perdido posteriormente al detectar resultados inconsistentes.

4) Generación automática de informes: Se pueden generar informes en *Microsoft Excel* utilizando tablas dinámicas (en inglés, *pivot tables*). Además se pueden exportar los resultados a los formatos estándar CSV [14] o JSON [1].

III. Aplicaciones prácticas

En esta sección se detallan dos ejemplos de aplicaciones prácticas, la primera por el autor del trabajo y la segunda por un equipo de programación competitiva. Por último, se detallan las futuras líneas de avance para el proyecto.

A. Experiencias de uso

El *framework* de desarrollo propuesto se ha utilizado para el desarrollo de un trabajo de investigación presentado a este mismo congreso, titulado "Un algoritmo eficiente para el problema de disposición de instalaciones en dos filas". En dicho trabajo se presentan resultados muy prometedores para el problema, empleado tiempos muy reducidos.

Por otra parte, es importante evaluar el uso de propuestas como *Mork* por terceras personas ajenas al proyecto. Con ese fin, se propuso su utilización al equipo de programación competitiva *Stackies* para el concurso de optimización anual *HashCode*, organizado por *Google*[1]. Dicha competición consiste en resolver en menos de 4 horas un problema de optimización NP-Duro, para el que es inviable encontrar soluciones exactas. La presión del tiempo hace que repartir el trabajo entre los miembros del equipo de forma eficiente sea de suma importancia.

Los resultados obtenidos superaron las expectativas, quedando en duodécima posición a nivel nacional, habiendo un total de 281 equipos nacionales. La retroalimentación del equipo fue positiva, recalcando especialmente las ventajas del diseño modular para dividir fácilmente el trabajo en equipo, y el ahorro de tiempo al poder reutilizar diferentes componentes existentes.

Entre las mejoras propuestas, algunas de ellas pendientes de implementación, se encuentran la creación de nuevas plantillas algorítmicas para flexibilizar aún más el uso del *framework*, y la generación semiautomática de experimentos para comparar propuestas algorítmicas.

B. Trabajos futuros

Como se ha mencionado anteriormente, el diseño modular permite integrar fácilmente nuevas componentes, especialmente nuevos algoritmos. Por ello, por una parte, se propone implementar nuevas estrategias metaheurísticas en la librería. Ejemplos de nuevas estrategias metaheurísticas pueden ser *Scatter Search* y *Path Relinking* [13]. Por otra parte, integrar librerías existentes, como jMetal [11], para diferentes algoritmos especializados, como los algoritmos poblacionales.

Otro de los aspectos mejorables es el sistema de experimentación. Actualmente, es el usuario el que define manualmente cada algoritmo con cada componente especificando cómo se realiza la comparativa entre los diferentes algoritmos, y cómo se definen los experimentos. Aunque permite un gran control, puede ser una tarea bastante tediosa. Por ello, se propone extender dicho sistema con plantillas de experimentos predefinidas, como podrían ser comparación de búsquedas

[1]Para más información sobre las reglas y restricciones del concurso, consultar: https://codingcompetitions.withgoogle.com/hashcode/.

locales, comparación de constructivos, etc. De esta forma, se puede realizar un proceso de experimentación preliminar semi-automático que evalúe la contribución de cada componente, minimizando el trabajo manual repetitivo.

Por otra parte, una de las claves de la investigación es obtener retroalimentación en el menor tiempo posible. Cuanto antes se obtienen los resultados antes se pueden analizar y sacar conclusiones de ellos. Con la implementación actual se puede ejecutar código de forma secuencial o paralela de forma automática. La idea consistiría en crear un nuevo ejecutor, distribuido y escalable de forma horizontal, que sea capaz de distribuir la carga en diversos nodos de cómputo para acelerar mucho más la ejecución de los experimentos. Dicho sistema podría ser compartido entre varios investigadores, para maximizar el uso de los recursos.

Por último, una limitación de alcance del proyecto es la aplicabilidad únicamente a problemas mono-objetivo. Aunque en principio podría aplicarse a problemas multi-objetivo, se considera que queda fuera del alcance del proyecto en su estado actual.

IV. Conclusiones

En este trabajo se presenta un *framework* de desarrollo denominado *Mork* que permite aplicar estrategias heurísticas y algoritmos metaheurísticos a problemas de optimización complejos, sean estos de minimización o maximización, independientemente del número de restricciones. El *framework* es modular, fácilmente extensible y extremadamente eficiente computacionalmente. Gracias a su uso, puede acelerarse el desarrollo de aproximaciones para problemas complejos sin sacrificar funcionalidad o rendimiento. Al ser fácilmente extensible, puede integrarse con librerías existentes de forma fácil e intuitiva. Además, la funcionalidad que incluye evita los errores más comunes. La propuesta presentada se ha aplicado con éxito a varios proyectos. Puede encontrarse el código y la documentación en el siguiente repositorio libre de *Github*: https://github.com/rmartinsanta/mork, ¡cualquier retroalimentación o contribución es bienvenida!

Bibliografía

[1] Tim Bray. *The JavaScript Object Notation (JSON) Data Interchange Format*. RFC 7159. Mar. 2014. DOI: 10.17487/RFC7159. URL: https://rfc-editor.org/rfc/rfc7159.txt.

[2] IBM ILOG Cplex. "V12. 1: User's Manual for CPLEX". In: *International Business Machines Corporation* 46.53 (2009), p. 157.

[3] Geoffrey De Smet and open source contributors. *OptaPlanner User Guide*. OptaPlanner is an open source constraint solver in Java. Red Hat, Inc. or third-party contributors. 2006. URL: https://www.optaplanner.org.

[4] Michel Gendreau, Jean-Yves Potvin, et al. *Handbook of metaheuristics*. Vol. 2. Springer, 2010.

[5] Fred W Glover and Gary A Kochenberger. *Handbook of metaheuristics*. Vol. 57. Springer Science & Business Media, 2006.

[6] LLC Gurobi Optimization. *Gurobi Optimizer Reference Manual*. 2021. URL: http://www.gurobi.com.

[7] Pierre Hansen and Nenad Mladenović. "Variable neighborhood search". In: *Handbook of metaheuristics*. Springer, 2003, pp. 145–184.

[8] Pierre Hansen and Nenad Mladenović. "Variable neighborhood search". In: *Search methodologies*. Springer, 2005, pp. 211–238.

[9] Douglas M Hawkins. "The problem of overfitting". In: *Journal of chemical information and computer sciences* 44.1 (2004), pp. 1–12.

[10] *MATLAB Optimization Toolbox*. The MathWorks, Natick, MA, USA.

[11] Antonio J. Nebro, Juan J. Durillo, and Matthieu Vergne. "Redesigning the JMetal Multi-Objective Optimization Framework". In: *Proceedings of the Companion Publication of the 2015 Annual Conference on Genetic and Evolutionary Computation*. GECCO Companion '15. Madrid, Spain: Association for Computing Machinery, 2015, pp. 1093–1100. ISBN: 9781450334884. DOI: 10.1145/2739482.2768462. URL: https://doi.org/10.1145/2739482.2768462.

[12] Mauricio GC Resende and CC Ribeiro. "Greedy randomized adaptive search procedures (GRASP)". In: *AT&T Labs Research Technical Report* 98.1 (1998), pp. 1–11.

[13] Mauricio GC Resende et al. "Scatter search and path-relinking: Fundamentals, advances, and applications". In: *Handbook of metaheuristics*. Springer, 2010, pp. 87–107.

[14] Yakov Shafranovich. *Common Format and MIME Type for Comma-Separated Values (CSV) Files*. RFC 4180. Oct. 2005. DOI: 10.17487/RFC4180. URL: https://rfc-editor.org/rfc/rfc4180.txt.

Índice de autores

A
Abad-Navarro, Francisco, 806
Afsar, Sezin, 465
Aguado, Esther, 116
Aguilera-Martos, Ignacio, 685
Aguirre, Aitor, 108
Alba, Enrique, 23, 219, 941, 947, 969, 970, 1013
Alberto da Silva Abreu, Luiz, 168
Alcalde, Cristina, 262, 305
Aledo, Juan A., 128, 247, 566
Alegre, Fernando, 989
Alfaro, Juan C., 128
Almagro Cádiz, Mario, 1026
Alonso, Maria Teresa, 629
Alonso, Pedro, 317
Alonso Ayuso, Antonio, 593
Alonso Betanzos, Amparo, 613, 735
Alonso-Moral, Jose M., 274, 298
Alonso-Rial, Adrián, 955
Alonso-Weber, Juan M., 217
Alvarado Díaz, Jorge, 505, 537, 1009
Álvarez, Verónica, 215
Alvarez-Valdes, Ramon, 629
Amaya, Jhon Edgar, 499
Andrés Arroyo, Marta, 235
Andreu-Vilarroig, Carlos, 566
Antunes-Santos, Felipe, 83
Aragón Royón, Francisco, 775
Araujo, Lourdes, 824
Arbelaitz, Olatz, 189, 818
Arias, Joaquín, 190
Arias-Rodil, Manuel, 685, 846
Arnaiz-González, Álvar, 691, 783, 785, 787
Arnaiz-Rodríguez, Adrián, 667
Arregui Pérez de Loza, Iñigo, 763
Arruti, Andoni, 201, 203
Arzamendia, Mario, 77
Asencio Martín, Lucía, 188
Atencia Payares, Luz Karime, 889
Aznar, Fidel, 102

B
Baena-García, Manuel, 648
Bahamonde, Antonio, 29, 735
Ban, Yue, 936
Barbudo Lunar, Rafael, 90
Barranco, Carlos D., 717
Barrenetxea Berasategi, Xabier, 180
Barri Khojasteh, Samad, 800
Bautista-Valhondo, Joaquín, 471
Baz, Juan, 331
Beltrán-Royo, César, 167
Benítez, José M., 775
Benítez-Caballero, María José, 393
Benito-Picazo, Jesús, 729
Bharatia, Harshal, 96
Bibiloni, Pedro, 211, 1031
Bidaurrazaga Barrueta, Arkaitz, 1062
Bielza, Concha, 840
Blanco-Mallo, Eva, 1041
Bobillo, Fernando, 351, 989
Bodi, Maria, 793
Bolón Canedo, Verónica, 735, 1041
Borrajo, Daniel, 186
Botana-López, Alberto, 955
Bueno, Sara, 1023
Bueno-Crespo, Andrés, 543
Bugarín-Diz, Alberto, 292, 298
Burusco, Ana, 262, 305
Bustince, Humberto, 223, 317, 358, 370, 376, 511, 613, 652

C
Cabrera, Inma P., 389
Cabrera, Marcelino, 434
Calleja, Pablo, 53
Calvo-Lluch, África, 723
Camacho, David, 523, 619
Camero, Andrés, 219
Campillo-Ageitos, Elena, 824
Campomanes-Álvarez, B. Rosario, 747
Campomanes-Álvarez, Carmen, 747
Campos, Manuel, 812
Campos Knupp, Diego, 168
Canabal-Juanatey, Mariña, 298
Cano Jiménez, Jesús, 889
Canovas-Segura, Bernardo, 812
Carabe, Luis, 209
Carmona, Cristobal J., 227, 231
Carmona, Enrique, 1023
Carmona, Javier, 446
Carmona del Jesús, Cristóbal José, 661
Carranza-García, Manuel, 658, 1091
Carrasco, Jacinto, 685
Carrasco-Villalón, Rocio, 717
Casado Ceballos, Alejandra, 472, 490
Casado-García, Ángela, 76, 134
Casado-Vara, Roberto, 712
Casanova, Jorge, 936
Casas-Martínez, Pedro, 490
Casas-Ortiz, Alberto, 997
Cascallar-Fuentes, Andrea, 292
Casillas, Jorge, 706
Castaño Rosa, Raúl, 919
Castejón, Federico, 1023
Castell Díaz, Javier, 806
Castellano-Quero, Manuel, 205
Castiblanco Ruiz, Fabian Alberto, 329
Castillo, Jhonny, 907
Castillo, Pedro, 505, 977
Castillo-López, Aitor, 376, 511
Castrillon-Santana, Modesto, 64
Castro, Jairo, 174
Castro, Javier, 241, 391
Catala, Alejandro, 298
Cavero Díaz, Sergio, 452, 1047
Ceberio Uribe, Josu, 453, 459
Cermeño, Eduardo, 209
Cernuda, Carlos, 108
Chanava, Walter, 757
Charte, David, 235

Charte, Francisco, 231
Chávez, Francisco, 505, 537
Chica, Manuel, 280
Chicano, Francisco, 941, 947
Cichos, Frank, 213
Cikel, Kevin, 77
Cintrano, Christian, 969, 970
Cisneros, Eusebio, 901
Cobo, Manuel Jesus, 337
Colmenar, José Manuel, 587, 1116
Colomine, Feijoo, 429, 517
Conti, Dante, 763
Corchado, Juan M., 712
Cordero, Pablo, 389
Cordón, Óscar, 180, 280, 601, 613, VII
Cornejo, M. Eugenia, 398
Corral de la Calle, Javier, 806
Correia da Silva Jardim, Lucas, 168
Cortés, Juan Carlos, 555, 566
Costa Cortez, Nahuel, 1110
Cotta, Carlos, 429, 499, 517
Cristóbal, Álvaro, 351
Cruz Corona, Carlos, 168
Cuadrado, David, 793
Cuesta-Infante, Alfredo, 114, 167
Cuesta Llavona, Cristina, 985

D
DAHI, Zakaria Abdelmoiz, 941, 947
de la Caba, Koro, 262
de La Cal, Enrique, 800
de la Morena Barrio, M. Eugenia, 806
de La Paz López, Félix, 1026
del Campo Ávila, José, 1079
Del Jesús Díaz, María José, 41, 227
Del Ser Lorente, Javier, 529
Delgado, Alvaro, 572
Delgado-Osuna, José Antonio, 753
Diaz, Irene, 192, 1017
Díaz, Josefa, 505
Díaz-Díaz, Norberto, 717
Diez, Francisco Javier, 221, 846
Díez, Jorge, 29, 741, 985
Diez-Olivan, Alberto, 180
Diosdado, Daniel, 166
Domínguez, César, 58, 89, 795
Dominguez, Enrique, 1053
Dominguez-Catena, Iris, 345
Dorado Llamas, Ignacio, 459
Duarte, Abraham, 452, 472, 601, 1047, 1097, 1103, 1116

E
E-Martín, Yolanda, 186
Echanobe, Javier, 931, 936
Elizondo, David, 231
Elorza, Anne, 1074
Escobar Mimbrera, Ángela, 41
Esnaola-Gonzalez, Iker, 221
Espínola, Rosa, 241, 391
Esteban Toscano, Aurora, 673
Estévez, Sonia, 679

F
Fáñez Kertelj, Mirko, 800
Fdez-Olivares, Juan, 115
Fedorova, Stanislava, 961
Fernández, Alberto, 235
Fernández, Guillermo, 247
Fernández, Javier, 223, 358, 376, 511, 613
Fernández Ares, Antonio, 977
Fernández-Breis, Jesualdo Tomás, 802, 806
Fernández de Vega, Francisco, 411, 505, 537, 1009
Fernandez-Izquierdo, Alba, 122
Fernández Leiva, Antonio J., 429, 499, 517
Fernández-Madrigal, Juan-Antonio, 205
Fernández Serrano, Ricardo, 153
Ferrer, Javier, 23, 1013
Ferrero-Jaurrieta, Mikel, 358
Fischer, Alexander, 213
Flaminio, Tommaso, 256
Flores Gallego, Maria Julia, 70
Fontenla-Romero, Oscar, 141
Franco De Los Ríos, Camilo, 329
Fresno Fernández, Víctor, 1026
Fuentes-Fernández, Rubén, 976
Fuentetaja, Raquel, 186
Fumanal Idocin, Javier, 370, 511

G
Gabilondo, Iñigo, 818
Galar, Mikel, 345
Galeano, Jesús, 446
Gallego-Fernández, Javier, 292
Gallego-Ledesma, Luis, 658
Gámez, José A., 128, 247, 640
Garanganga, Bradwell, 895
Garcia-Andoin, Mikel, 931
García-Aragón, Roberto, 397
García Arenas, Maribel, 977
García-Barzana, Marta, 685
García-Castro, Raúl, 122
García-Cerezo, Alfonso, 205
García-Domínguez, Manuel, 76, 795
García-Gil, Diego, 685
García Gómez, Pablo, 147
García-González, Jorge, 729
García-Gutiérrez, Jorge, 1091
García Herrero, Jesús, 881
García-Magariño, Iván, 976
García-Martínez, Carlos, 753
García-Osorio, César, 783, 787
García Pardo, Eduardo, 452, 490, 581, 593, 1047
García-Pedrero, Angel, 836
Garcia-Retuerta, David, 712
García-Rodríguez, Sara, 1017
García Sánchez, Pablo, 499, 977
Garcia-Vico, Angel M., 227, 231
García-Zamora, Diego, 323
Garciarena, Unai, 405
Garitano, Iñaki, 108
Garmendia, Luis, 364
Garnica, Oscar, 153, 549
Garrido, Roberto, 834
Garrido-Labrador, José Luis, 667
Garrido-Merchán, Eduardo C., 159, 188, 703
Garrocho, Alejandro, 174
Gatt, Albert, 274
Gauchola, Jaume, 832
Gayoso Heredia, Marta, 923

Gibaja, Eva, 697
Gibert, Karina, 763
Gil Borrás, Sergio, 593
Gil-Merino, Rodrigo, 941, 947
Giménez-Palacios, Iván, 629
Giménez-Solano, Vicente Miguel, 802
Giráldez-Cru, Jesús, 280
Godo, Lluis, 256
Gomez, Daniel, 241, 364
Gómez, Daniel, 391
Gomez, Fernando, 174
Gomez, Josep, 793
Gómez, Juan Antonio, 913
Gómez, Marisol, 370
Gómez-Sanz, Jorge J., 976
Gómez-Valadés, Alba, 838
González, Antonio, 268
Gonzalez, Víctor, 800
González-Briones, Alfonso, 976
González-Carvajal, Santiago, 703
González Doncel, Gaspar, 153
González García, Pedro, 41, 227, 231, 661
González-Hidalgo, Manuel, 211, 1031
González-Moya, Francisco Javier, 691
González Naharro, Luis, 70
Gonzalez-Pardo, Antonio, 607
González-Pérez, Beatriz, 560
González-Reina, Adán, 723
González Rodríguez, Inés, 147
Gonzalez-Sanchez, Belen, 160, 478
Gonzalo Martin, Consuelo, 834, 836
Gordo-Martín, Daniel, 955
Gragera, Alba, 186
Gragera, Miguel Ángel, 543
Gregor, Derlis, 77
Guerrero, Juan José, 23, 1013
Guerrero, Pedro, 262
Guevara, Juan Antonio, 391
Guijarro-Berdiñas, Bertha, 141
Guijo-Rubio, David, 641, 660
Gurrutxaga, Ibai, 818
Gutiérrez, Alberto, 549
Gutiérrez, Inmaculada, 241, 391
Gutierrez, Juan Carlos, 174
Gutierrez, Nuria, 834
Gutiérrez, Pedro Antonio, 641, 660
Gutiérrez-Avilés, David, 639
Gutiérrez Reina, Daniel, 5, 11, 17, 77

H
Halodova, Patricie, 153
Hemberg, Erik, 233
Heras, Jónathan, 58, 76, 89, 134, 795
Heredia Vizcaino, Diana, 907
Hermoso Peralo, Roberto, 889
Hernández-Díaz, Alfredo G., 627
Hernández-García, Sergio, 114
Hernando, Leticia, 1074
Herrán, Alberto, 587
Herrera, Francisco, 225, 235, 280, 529, 652, 685
Hervás-Martínez, César, 641, 660
Hidalgo, José Ignacio, 153, 549, 572
Holubek, Viktor, 213
Huertas-García, Álvaro, 619
Huertas-Tato, Javier, 523, 619
Huidobro, Pedro, 317
Huitzil, Ignacio, 351, 989

I
Idocin, Javier, 613
Iglesias, Natalia, 812
Iglesias Martínez, José Antonio, 923
Iglesias-Rey, Sara, 83
Inés Armas, Adrián, 58, 76
Irazabal-Urrutia, Oier, 189
Irigoyen, Eloy, 203
Iturbe, Mikel, 108

J
Jabba, Daladier, 907
Janis, Vladimír, 317
Jara, Leonardo, 268
Jariego Pérez, Luis C., 159
Jelić, Marko, 221
Jiménez, Daniel, 453
Jiménez Agudelo, Yury Andrea, 757
Jimenez Linares, Luis, 286
Jiménez Merino, Ernesto, 593
Jiménez-Navarro, M. J., 659
Jin, Yaochu, VIII
José da Silva Neto, Antônio, 168
Juarez, Jose M., 812
Juez-Gil, Mario, 783, 785, 787, 993

K
Kronberger, Gabriel, 153

L
Labella Romero, Álvaro, 323, 385, 1037
Lafuente, Julio, 652
Lanchares, Juan, 549
Lara-Benítez, Pedro, 658
Larrañaga, Pedro, 840
Leceta, Itsaso, 262
Ledezma Espino, Agapito, 869, 913, 919, 923
Legaz-García, María del Carmen, 802
Lillo-Saavedra, Mario, 836
Llamazares Rodríguez, Bonifacio, 357
Llerena Caña, Juan Pedro, 881
Llopis-Ibor, Laura, 167
Lobo, David, 398
López, Beatriz, 832
López, David, 685
Lopez, Samuel, 174
Lopez, Victoria, 679, 757
Lopez-Molina, Carlos, 83, 376
López-Navarro, Elena, 555
López-Nozal, Carlos, 783, 785, 993
López-Riobóo Botana, Iñigo Luis, 735
López-Rubio, Ezequiel, 729
López-Sánchez, Ana Dolores, 627
López Vargas, Ascension, 919
Lorenzo-Navarro, Javier, 64
Losada Casado, Adrián, 881
Lozano, José Antonio, 215, 453, 1074, 1085, 1106
Lozano-Osorio, Isaac, 601, 1103
Luaces, Oscar, 29
Luengo, Julián, 685
Luna, Francisco, 446

Luna, Jose Maria, 769
Luna-Romera, José M., 658
Luque, Gabriel, 941, 947
Luque, Manuel, 846
Luque-Baena, Rafael M., 1001, 1005
Luzón, María V., 225

M
Madrid, Nicolas, 395
Magan Lopez, Elena, 875
Magdalena, Luis, 364
Maldonado, José Alberto, 802
Mandow, Lawrence, 135, 187
Marcos, Mar, 802
María-Armingol, José, 857, 863
Marín-Plaza, Pablo, 857, 863
Mariotti, Ettore, 274
Martí, Rafael, 631, 633
Martín, Alejandro, 523, 619
Martin, Jose Ignacio, 201, 203
Martín-Albo, Sergio, 135
Martín Chozas, Patricia, 47, 53
Martín Santamaría, Raúl, 587, 1116
Martínez, Aritz, 529
Martínez, Luis, 323, 385, 1037
Martínez, María, 331
Martínez-Álvarez, F., 659
Martínez-Cámara, Eugenio, 225
Martínez-Cendán, Juan-Pedro, 543
Martínez Crespo, Jorge, 919
Martínez-Eguiluz, Maitane, 189, 818
Martínez España, Raquel, 543
Martínez-Gavara, Anna, 627, 631, 633
Martínez-Más, José, 543
Martinez Rodriguez, Raquel, 201, 203
Martinez-Romo, Juan, 824
Martínez-Salvador, Begoña, 802
Martinez-Sanchez, Antonio, 207
Martínez Tomás, Rafael, 838
Massana, Joaquim, 832
Massanet, Sebastia, 223
Mata, Eloy Javier, 58, 89, 795
Mazuelas, Santiago, 215
Medina, Jesús, 393, 397, 398
Medina Burga, Melissa, 901
Melgar-García, Laura, 639
Menasalvas Ruiz, Ernestina, 804, 834
Mendiburu, Alexander, 405
Mercado Palomino, Elia, 775
Merino, María, 1106
Miguel, Diego, 993
Millán, Laura, 153
Miñana, Juan José, 304, 311
Mir-Fuentes, Arnau, 83
Molina-Cabello, Miguel A., 1001, 1005
Molina Cabrera, Daniel, 529
Molina López, José Manuel, 881
Montañana, Ricardo, 640
Montero, Javier, 329, 364
Montes, Susana, 253, 317, 331, 1017
Montiel-Ponsoda, Elena, 47, 53
Mora, Antonio, 977
Mora López, Llanos, 1079
Morales, Antonio, 812
Morales-Bueno, Rafael, 648
Morell Martínez, José Ángel, 970
Moreno Garcia, Juan, 286
Moreno-Rebato, Mar, 190
Morita Hernández, Martín, 411, 1009
Moscardó-García, Ana, 555
Móstoles Rodríguez, Roberto, 881
Muguerza, Javier, 201, 203, 818
Muiños-Landín, Santiago, 213, 955
Muñoz, Andres, 543
Murombedzi, James, 895
Murueta-Goyena, Ane, 818

N
Nesmachnow, Sergio, 191
Nikolov-Vasilev, Vladislav, 115
Novoa-Hernández, Pavel, 484
Novoa-Paradela, David, 141
Nuñez, Concepción, 560
Núñez-Molina, Carlos, 115
Núñez-Peiró, Miguel, 923
Nyakutambwa, Trymore, 895

O
Ojeda-Aciego, Manuel, 395
Ojeda-Hernandez, Manuel, 389, 1059
Olivares-Gil, Alicia, 667
Onaindia, Eva, 166
O'Reilly, Una-May, 233
Orellana, Aitor Manuel, 337
Orengo, Juan C., 566
Ortegón, Juan Carlos, 434
Ortiz, Alberto, 304, 311
Ortiz, Esaú, 304, 311
Ortiz-González, Ana, 543
Ortiz-Reina, Sebastián, 543
Ortiz-Toro, César Antonio, 834, 836
Ossowski, Sascha, 190
Otero-Tranchero, Jacobo, 955
Ottonello, Nicolás, 191

P
Pacheco, Santiago, 191
Palacios, Juan Jose, 465
Pantrigo, Juan José, 167
Parra, Daniel, 549
Parreño, Francisco, 629
Pascual, Vico, 58, 89, 795
Pascual Triana, José Daniel, 235
Paternain, Daniel, 345
Pavón, Juan, 976
Pedraza, Tatiana, 387
Pelta, David, 434, 484
Peralta, Federico, 5, 11, 17
Peregrin, Antonio, 174, 417
Pereira Dimuro, Graçaliz, 358, 370
Perez, Aritz, 1062
Pérez, Carlos J., 423
Perez, Ignacio Javier, 337
Pérez, Raúl, 115, 268
Pérez-Asurmendi, Patrizia, 357
Pérez de la Cruz, José Luis, 135, 187
Perez-Fernandez, Raul, 331
Pérez-Martín, Jorge, 846
Pérez Martos, Luis Alfonso, 661

Pérez-Núñez, Pablo, 29, 741, 985
Pérez-Peló, Sergio, 472, 607, 633, 1097
Perona, Iñigo, 818
Pineda-Palencia, Ismael, 1001
Pineda Torres, Ivo, 851
Pinheiro Domingos, Roberto, 168
Porras Castaño, Javier, 1068
Poveda-Villalón, María, 122
Poyatos Amador, Javier, 529
Pozanco, Alberto, 186
Prieto Santamaría, Lucía, 804
Puerta Callejón, José Miguel, 70, 247, 640
Pujić, Dea, 221
Pulido, Inmaculada, 174

Q
Quiñones, Alejandro, 446
Quirós, Pelayo, 747

R
R. Vela, Camino, 147, 192, 465
Rais, Jasmina, 102
Ramírez, Aurora, 35
Ramírez Poussa, Eloisa, 393, 397
Ramírez-Sanz, José Miguel, 667
Ramos-Jiménez, Gonzalo, 729
Ramos-Soto, Alejandro, 292
Ranchal-Parrado, Daniel, 753
Raya, Òscar, 832
Reguera-Bakhache, Daniel, 108
Remeseiro, Beatriz, 1041
Remezal-Solano, Manuel, 543
Riaño, David, 793
Riaño Sánchez, Miguel A, 840
Rico, Noelia, 192, 1017
Rico Prieto, Iván, 889
Riera, Juan Vicente, 223
Rincón, Mariano, 838
Riquelme, José C., 1091
Rodríguez, Alejandro, 793
Rodriguez, Álvaro, 17
Rodríguez, J. Tinguaro, 329
Rodríguez, Juan J., 691, 783, 785, 787
Rodriguez, Miguel Angel, 417
Rodríguez, Rosa M., 323, 385, 1037
Rodríguez-Barroso, Nuria, 225
Rodriguez-Benitez, Luis, 286
Rodriguez Doncel, Victor, 53
Rodríguez-García, José A., 190
Rodríguez Gómez, Francisco, 1079
Rodríguez González, Alejandro, 804, 834
Rodríguez-López, Jesús, 387
Rodríguez-Martínez, Iosu, 358, 652
Rodríguez-Revuelta, Antonio, 741
Roiz Pagador, Joaquín, 723
Rojas Ramos, Fernando, 851
Romero, Alberto, 166
Romero, José Raúl, 35, 90
Rossi, Claudio, 116
Royo, Didac, 76
Rubio-Escudero, Cristina, 639
Rubio Perona, Fernando, 70
Ruiz, Roberto, 723
Ruiz-Casado, Jose Luis, 1005
Ruiz-de-la-Cuadra, Alejandro, 857, 863
Ruiz-Rivas Hernando, Ulpiano, 919
Rus Rus, Rafael, 869
Rutabingwa, Frank, 895

S
Saborido, Rubén, 23, 229, 1013
Said Hung, Elías, 907
Salamanca, Juan Jesús, 253
Salazar-Ramirez, Asier, 201, 203
Salerno, Mauricio, 186
Sánchez, Almudena, 537
Sanchez, Arturo, 901
Sánchez, José L., 560
Sanchez-Gomez, Jesus M., 423
Sánchez-Guevara Sánchez, Carmen, 923
Sánchez-Nielsen, Elena, 64
Sánchez-Oro, Jesús, 472, 490, 1097, 1103
Sánchez Ramos, Luciano, 1110
Sanchis, Araceli, 217, 875
Sanmartin, Paul, 907
Santamaría, Gonzalo, 89
Santana, Roberto, 405, 642, 1062, 1085
Santander-Jiménez, Sergio, 160, 440, 478
Santos, Olga C., 997
Sanz, Ricardo, 116
Sanz-Fernández, Ana, 923
Sarmiento-Ramírez, Jonay, 64
Saunders, Anthony, 292
Sedano, Javier, 800
Sedova, Ekaterina, 187
Seker, Huseyin, 227
Sesma, Mikel, 376, 652
Sesmero Lorente, María Paz, 217, 875
Sierra, Basilio, 201
Sierra, Ignacio, 717
Simmonds, Jose, 913
Slezak, Dominik, 393
Sousa, Leonel, 440

T
Tabik, Siham, 529
Taillefer, Eugenia, 648
Talavera-Martínez, Lidia, 211, 1031
Tapia, Alejandro, 17
Tena, Félix, 572
Teran, Diego, 1053
Tomasevic, Nikola, 221
Toral, Sergio, 5, 11, 17, 77
Torres, J. F., 659
Torrontegui, Erik, 936
Toutouh, Jamal, 219, 233, 969, 970
Trasierras, Antonio Manuel, 769
Troncoso, Alicia, 639, 659

U
Unanue, Imanol, 1106
Urdanpilleta, Marta, 262
Ureña Espa, Aurelio, 775
Urio, Asier, 370
Ursúa, Pablo, 652

V
Vadillo, Jon, 642, 1085
Valenzuela, Juan Francisco, 446
Valero, Óscar, 304, 311, 387

Valverde, Gabriel, 560
Varela, Alejandro, 572
Vargas, Víctor Manuel, 641, 660
Vázquez-Flores, Karen Leticia, 47
Vega-Rodríguez, Miguel A., 160, 423, 440, 478
Velasco, José Manuel, 537, 560, 572
Velez-Estevez, Antonio, 337
Vellido, Alfredo, 830
Vellido, Ignacio, 115
Ventura, Sebastián, 35, 90, 673, 753, 769
Vidaurre, Carmen, 370
Villanueva, Rafael-J., 555, 566
Villar, David, 706
Villar, José Ramón, 800
Villar-Rodríguez, Guillermo, 523
Villegas Cortez, Juan, 411
Villota, María, 89

Y
Yagüe-Cuevas, David, 857, 863
Yanes, Samuel, 5, 11, 17
Yuste, Javier, 581

Z
Zafra Gómez, Amelia, 673, 697
Zapata, Miguel Ángel, 76
Zapata, Pablo, 446
Zurutuza, Urko, 108